IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:	**Philyaw et al.**	**Confirmation No.:**	**8055**
Serial No.:	**09/417,863**		
Filed:	**October 13, 1999**		
Group:	**2142**		
Examiner:	**Douglas B. Blair**		
For:	**SOFTWARE DOWNLOADING USING A TELEVISION BROADCAST CHANNEL**		

BRIEF ON APPEAL

Gregory M. Howison
HOWISON & ARNOTT, L.L.P.
Attorneys for Appellants
P.O. Box 741715
Dallas, Texas 75374-1715
Phone: (972) 680-6050
Facsimile: (972)-479-0462
e-mail: patents@dalpat.com

TABLE OF CONTENTS

Appendices:

TABLE OF AUTHORITIES

Cases

Rules

Regulations

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:	**Philyaw et al.**	**Confirmation No.:**	**8055**
Serial No.:	**09/417,863**		
Filed:	**October 13, 1999**		
Group:	**2142**		
Examiner:	**Douglas B. Blair**		
For:	**SOFTWARE DOWNLOADING USING A TELEVISION BROADCAST CHANNEL**		

APPELLANTS' MAIN BRIEF ON APPEAL

This Brief is submitted in accordance with 37 C.F.R. § 41.67 concerning the Notice of Appeal filed February 28, 2007 in response to the Final Office Action dated August 8, 2006, wherein the Examiner finally rejected claims 1-30 that comprise all of the pending claims in this application.

I. Real Party Interest.

The party in interest is L.V. Partners, L.P., a Texas limited partnership, whose general partner is LV GP, L.L.C., and whose principal office and place of business is at 2626 Cole Avenue, Dallas, Texas 75204.

II. Related Appeals and Interferences.

Appellants have the following related application pending appeals:

- U.S. Patent Application Serial No. 07/614,937, Appeal No. 2007-1745 entitled "LAUNCHING A WEB SITE USING A PASSIVE TRANSPONDER" (Atty. Dkt. No. PHLY-25,356), filed on July 11, 2000;

- U.S. Patent Application Serial No. 09/494,924 entitled "INPUT DEVICE FOR ALLOWING INTERFACE TO A WEB SITE IN ASSOCIATION WITH A UNIQUE INPUT CODE" (Atty. Dkt. No. PHLY-24,913), filed on February 1, 2000;

- U.S. Patent Application Serial No. 10/884,377 entitled "OPTICAL READER WITH ULTRAVIOLET WAVELENGTH" (Atty. Dkt. No. PHLY-26,826) filed on July 2, 2004; and

- U.S. Patent Application Serial No. 09/382,421 entitled "COMBINED PRODUCT CODE AND INSIGNIA FOR SIGNIFYING AN INTERNAL INTERACTIVE CODE" (Atty. Dkt. PHLY-24,740) filed on August 24, 1999.

Appellants have filed Notices of Appeal in the following related applications:

- U.S. Patent Application Serial No. 09/659,520 entitled "LAUNCHING A WEB SITE USING A PERSONAL DEVICE" (Atty. Dkt. No. PHLY-25,355), filed on September 12, 2000;

- U.S. Patent Application Serial No. 09/382,423 entitled "METHOD AND APPARATUS FOR UTILIZING AN AUDIBLE SIGNAL TO INDUCE A USER TO SELECT AN E-COMMERCE FUNCTION" (Atty. Dkt. No. PHLY-24,739), filed on August 24, 1999;

- U.S. Patent Application Serial No. 09/382,374 entitled "METHOD AND APPARATUS FOR ALLOWING A BROADCAST TO REMOTELY CONTROL A COMPUTER" (Atty. Dkt. No. PHLY-24,736), filed on August 24, 1999;

- U.S. Patent Application Serial No. 09/659,170 entitled "ACCESSING A VENDOR WEB SITE USING PERSONAL ACCOUNT INFORMATION RETRIEVED FROM A CREDIT CARD COMPANY WEB SITE" (Atty. Dkt. No. PHLY-25,340), filed on September 11, 2000;

- U.S. Patent Application Serial No. 09/602,034 entitled "CONTROLLING A PC USING A TONE FROM A CELLULAR TELEPHONE" (Atty. Dkt. No. PHLY-25,337), filed on June 23, 2000;

- U.S. Patent Application Serial No. 09/382,372 entitled "METHOD AND APPARATUS FOR MATCHING A USER'S USE PROFILE IN COMMERCE WITH A BROADCAST" (Atty. Dkt. No. PHLY-24,738), filed August 24, 1999;

- U.S. Patent Application Serial No. 09/642,891 entitled "RETRIEVING PERSONAL ACCOUNT INFORMATION FROM A WEB SITE BY READING A CREDIT CARD" (Atty. Dkt. No. PHLY-25,338), filed on August 21, 2000.

The above-identified patent application has no related interferences.

III. Status of the Claims.

Claims 1-30 from the application are pending, stand firmly rejected, and are on appeal here. A complete and current listing of Claims 1-30 are attached here in the **CLAIMS APPENDIX**.

IV. Status of Amendments.

Appellants filed an Amendment and Response to Office Action on August 7, 2006 in response to the Office Action, mailed July 5, 2006. This Amendment and Response to Office Action was the last Response amending claims.

V. Summary of the Claimed Subject Matter.

The present invention, as set forth currently in independent Claim 1, relates to a method for distributing software. The method comprises the step of providing a television broadcast distribution system[1] having one or more broadcast channels[2] for broadcasting analog and digital[3] television information to a receiver[4] of a user.[5] The method further comprises the step of

[1] See Specification page 30, lines 23-24.
[2] See Specification page 31, lines 10-13; page 34, lines 15-18.
[3] See Specification page 31, lines 7-9.
[4] See Specification Reference # 1608 on Figure 16; page 31, lines 2-7.
[5] See Specification Figure 1; page 9, lines 1-25; page 12, lines 1-4; page 13, lines 1-7.

designating select ones of the one or more broadcast channels for the transmission of one or more discrete software data streams.[6] The method further comprises the step of transmitting the one or more discrete software data streams over the select ones of the one or more broadcast channels[7] at a scheduled time,[8] each of the one or more discrete software data streams having a unique ID associated therewith,[9] which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user.[10] The method further comprises the step of the user associating with a monitoring interface[11] of the user that is connected to the receiver, the unique ID associated with that user and a desired one of the one or more discrete software streams.[12] The method further comprises the step of selecting, in the monitoring interface associated with the user, selected ones of the one or more discrete software data streams according to the respective unique IDs for download via the receiver.[13] The method further comprises the step of downloading the selected one or more discrete software data streams to a user storage device during the scheduled time for use by the user, the user storage device connected to the receiver through said monitoring interface.[14] The method further comprises the step of deleting the unique ID from the monitoring interface for each of the one or more discrete software data streams downloaded after downloading thereof.[15]

The present invention, as set forth currently in dependent Claim 2, relates to the method of Claim 1, where the television broadcast distribution system is a cable television broadcast system.[16]

The present invention, as set forth currently in dependent Claim 3, relates to the method of Claim 1 that further comprising an error checking step during the step of downloading such

---

[6] See Specification page 31, lines 10-25; page 34, lines 15-26.
[7] See Specification page 31, line 22 – page 32, line 4; page 32, lines 13-16.
[8] See Specification page 32, lines 16-18; page 34, lines 15-26; page 35, line 19 – page 36, line 14; page 38, lines 19-25.
[9] See Specification page 32, lines 4-18.
[10] See Specification page 42, lines 13-25.
[11] See Specification page 32, lines 23-25.
[12] See Specification page 42, lines 13-25.
[13] See Specification page 31, line 10 – page 33, line 13; page 34, line 15 – page 35, line 16.
[14] See Specification page 34, line 15 – page 35, line 16; page 39, lines 11-24.
[15] See Specification page 40, lines 11-21; page 42, lines 13-15; page 43, lines 17-19.
[16] See Specification page 31, lines 4-5.

that the one or more software data streams which are downloaded are checked for errors during the download process.[17]

The present invention, as set forth currently in dependent Claim 4, relates to the method of Claim 3, where the receiver automatically re-selects for download, and downloads, the one or more software data streams which fail the step of error checking.[18]

The present invention, as set forth currently in dependent Claim 5, relates to the method of Claim 1, where the step of broadcasting broadcasts the one or more software data streams repetitively during a specific period of time.[19]

The present invention, as set forth currently in dependent Claim 6, relates to the method of Claim 1, where the step of broadcasting broadcasts the one or more software data streams once during a specific time period.[20]

The present invention, as set forth currently in dependent Claim 7, relates to the method of Claim 1 that further comprising a step of accounting which logs the unique IDs of the one or more software data streams which were downloaded with an accounting device, and transmits the unique IDs to a provider of the one or more software data streams using the accounting device.[21]

The present invention, as set forth currently in dependent Claim 8, relates to the method of Claim 7, where the accounting device interfaces to a public-switched telephone network, and transmits the unique IDs over the public-switched telephone network to the provider of the one or more software data streams.[22]

The present invention, as set forth currently in dependent Claim 9, relates to the method of Claim 7, where the accounting device interfaces to a packet-switched global communication

[17] See Specification page 35, lines 7-18; page 40, lines 1-14.
[18] See Specification page 38, lines 10-16.
[19] See Specification page 37, line 9 – page 38, line 25.
[20] See Specification page 39, line 11 – page 40, line 24.
[21] See Specification page 33, lines 14-26; page 41, lines 13-23.
[22] See Specification page 33, lines 16-22.

network, and transmits the unique IDs over the global communication network to the provider of the one or more software data streams.[23]

The present invention, as set forth currently in dependent Claim 10, relates to the method of Claim 1, where the step of selecting further comprises the step of programming the receiver with time, channel, and the unique ID information.[24]

The present invention, as set forth currently in dependent Claim 11, relates to the method of Claim 1, where the one or more software data streams comprise software applications which are broadcast on a first channel, and one or more software updates which are broadcast on a second channel.[25]

The present invention, as set forth currently in dependent Claim 12, relates to the method of Claim 1, where the step of downloading downloads the selects ones of the one or more software data streams directly to a user computer over a communication link existing between the receiver and the user computer.[26]

The present invention, as set forth currently in dependent Claim 13, relates to the method of Claim 12, where the communication link is a universal serial bus.[27]

The present invention, as set forth currently in dependent Claim 14, relates to the method of Claim 12, where the communication link is a high-performance serial bus.[28]

The present invention, as set forth currently in dependent Claim 15, relates to the method of Claim 1, where the television broadcast distribution system is a digital television broadcast system.[29]

The present invention, as set forth currently in independent Claim 16, relates to a system for distributing software. The system further comprises a television broadcast distribution

---

[23] See Specification page 33, lines 22-26.
[24] See Specification page 32, lines 13-18; page 33, lines 6-9.
[25] See Specification page 33, lines 9-13.
[26] See Specification page 32, line 19 – page 33, line 13.
[27] See Specification page 33, lines 1-3.
[28] See Specification page 33, lines 1-3.
[29] See Specification page 31, lines 7-9.

system[30] having one or more broadcast channels[31] for broadcasting analog and digital[32] television information to a receiver of a user.[33] The system further comprises one or more discrete software data streams designated for transmission[34] on select ones of said one or more broadcast channels,[35] each of the one or more discrete data streams having a unique ID associated therewith,[36] which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user.[37] The system further comprises a user storage device[38] associated with the user and connected to said receiver through a monitoring interface[39] and said user storage device operable to store both one or more of the unique IDs associated with the user prior to downloading of the one or more discrete software data streams and, after downloading thereof, for storing said downloaded one or more discrete software data streams wherein each of said unique IDs stored in said user interface is stored there by the user.[40] The system further comprises wherein said one or more discrete software data streams are transmitted over said select ones of said one or more broadcast channels at a scheduled time, each of said one or more discrete software data streams having associated therewith the associated one of said unique IDs.[41] The system further comprises wherein said select ones of said one or more discrete software data streams are downloaded via said receiver to said monitoring interface for filtering[42] said discrete software data streams according to said respective unique IDs,[43] and wherein said unique ID for each of the one or more discrete software data streams downloaded deleted from the user storage device after downloading thereof.[44]

---

[30] See Specification page 30, lines 203-24.
[31] See Specification page 31, lines 10-13; page 34, lines 15-18.
[32] See Specification page 31, lines 7-9.
[33] See Specification page 31, lines 2-7; Reference #1608 on Figure 16; Figure 1; page 9, lines 1-25; page 12, lines 1-4; page 13, lines 1-7
[34] See Specification page 31, lines 10-24; page 34, lines 15-26.
[35] See Specification page 31, line 22 – page 32, line 4; page 32, lines 13-16.
[36] See Specification page 32, lines 4-18.
[37] See Specification page 32, lines 4-18; page 41, line 24 – page 42, line 25.
[38] See Specification page 32, lines 19-20.
[39] See Specification page 32, lines 23-25.
[40] See Specification page 32, lines 13-20.
[41] See Specification page 34, line 15 – page 35, line 6; page 39, lines 11-24.
[42] See Specification page 31, line 10 – page 32, line 20; page 34, lines 15-26.
[43] See Specification page 34, lines 24-26.
[44] See Specification page 40, lines 11-21; page 42, lines 13-15; page 43, lines 17-19.

The present invention, as set forth currently in dependent Claim 17, relates to the system of Claim 16, where the television broadcast distribution system is a cable television broadcast system.[45]

The present invention, as set forth currently in dependent Claim 18, relates to the system of Claim 16, where the select ones of said one or more software data streams are checked for errors when being downloaded.[46]

The present invention, as set forth currently in dependent Claim 19, relates to the system of Claim 18, where the receiver automatically re-selects for download, and downloads, said one or more software data streams which fail the error checking process.[47]

The present invention, as set forth currently in dependent Claim 20, relates to the system of Claim 16, where the one or more software data streams are broadcast repetitively during a specific period of time.[48]

The present invention, as set forth currently in dependent Claim 21, relates to the system of Claim 16, where the one or more software data streams are broadcast once during a specific time period.[49]

The present invention, as set forth currently in dependent Claim 22, relates to the system of Claim 16, where an accounting device logs said unique IDs of said one or more software data streams which were downloaded, and transmits said unique IDs to a provider of said one or more software data streams using said accounting device.[50]

The present invention, as set forth currently in dependent Claim 23, relates to the system of Claim 22, where the accounting device interfaces to a public-switched telephone network, and

[45] See Specification page 31, lines 4-5.
[46] See Specification page 35, lines 7-18; page 40, lines 1-14.
[47] See Specification page 38, lines 10-16.
[48] See Specification page 37, line 9 – page 38, line 25.
[49] See Specification page 39, line 11 – page 40, line 25.
[50] See Specification page 33, lines 14-26; page 41, lines 13-23.

transmits said unique IDs over said public-switched telephone network to said provider of said one or more software data streams.[51]

The present invention, as set forth currently in dependent Claim 24, relates to the system of Claim 22, where the accounting device interfaces to a packet-switched global communication network, and transmits said unique IDs over said global communication network to said provider of said one or more software data streams.[52]

The present invention, as set forth currently in dependent Claim 25, relates to the system of Claim 16, where the receiver is programmed by inputting parameters which comprise time, channel, and said unique ID information.[53]

The present invention, as set forth currently in dependent Claim 26, relates to the system of Claim 16, where the one or more software data streams comprise software applications which are broadcast on a first channel, and one or more software updates which are broadcast on a second channel.[54]

The present invention, as set forth currently in dependent Claim 27, relates to the system of Claim 16, where the select ones of said one or more software data streams are downloaded directly to a user computer over a communication link existing between said receiver and said user computer.[55]

The present invention, as set forth currently in dependent Claim 28, relates to the system of Claim 27, where the communication link is a universal serial bus.[56]

The present invention, as set forth currently in dependent Claim 29, relates to the system of Claim 27, where the communication link is a high-performance serial bus.[57]

---

[51] See Specification page 33, lines 16-22.
[52] See Specification page 33, lines 22-26.
[53] See Specification page 32, lines 13-18; page 33, lines 6-9.
[54] See Specification page 33, lines 9-13.
[55] See Specification page 32, line 19 – page 33, line 13.
[56] See Specification page 33, lines 1-3.
[57] See Specification page 33, lines 1-3.

The present invention, as set forth currently in dependent Claim 30, relates to the system of Claim 16, wherein said television broadcast distribution system is a digital television broadcast system.[58]

VI. Grounds of Rejection to be Reviewed on Appeal.

Claims 1-2, 5, 10-11, 15-17, 20, 25-26 and 30 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") in view of U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*"). Claims 3-4 and 18-19 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") in view of U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*"), and further in view of U.S. Patent to 5,666,293 to Metz et al. ("*Metz*"). Claims 6, 12-14, 21, and 27-29 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") in view of U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*"), and further in view of U.S. Patent 5,894,516 to Brandenburg ("*Brandenburg*"). Claims 7-8 and 22-23 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") in view of U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*"), and further in view of U.S. Patent to 5,003,384 to Durden. ("*Durden*"). Claims 9 and 24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") in view of U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*"), and U.S. Patent to 5,003,384 to Durden. ("*Durden*") and further in view of U.S. Patent to 6,317,885 to Fries. ("*Fries*").

As detailed below, Appellants believe that the Examiner has improperly applied the combination of the *Tarr* and *Birdwell* references to Claims 1-2, 5, 10-11, 15-17, 20, 25-26 and 30. Additionally, Appellants believe that the Examiner has improperly applied the combination of the *Tarr, Birdwell* and *Metz* references to Claims 3-4 and 18-19. Further, Appellants believe that the Examiner has improperly applied the combination of the *Tarr, Birdwell* and *Brandenburg* references to Claims 6, 12-14, 21 and 27-29. Appellants believe that the Examiner has improperly applied the combination of the *Tarr, Birdwell* and *Durden* references to Claims 7-8 and 22-23. Further, Appellants believe that the Examiner has improperly applied the combination of the *Tarr, Birdwell*, *Durden* and *Fries* references to Claims 9 and 24. Specifically, Appellants submit that the §103 rejections based on these combinations are not

[58] See Specification page 31, lines 7-9.

proper and are without basis, and that the Examiner has failed to state a *prima facie* case as to either combination constituting a viable combination of references under 35 U.S.C. § 103.

VII. Argument and Discussion.

In order to prevail, Appellants must show that Examiner has improperly combined *Tarr, Birdwell*, *Metz*, *Brandenburg*, *Durden* and *Fries* in support of the 35 U.S.C. § 103. As such, a brief discussion of the relevant rules and recent court decisions affecting a proper rejection under 35 U.S.C. § 103 follows.

A. Rejections under 35 U.S.C. §103

MPEP § 2142 specifies that:

> The examiner bears the initial burden of factually supporting any *prima facie* conclusion of obviousness. If the examiner does not produce a *prima facie* case, the applicant is under no obligation to submit evidence of nonobviousness.

In regard to what an examiner must show in order to establish a *prima facie* case of obviousness, MPEP § 2142 further explains that:

> To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. . . . Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

In regard to what an examiner must do in order to meet the first criterion for a prima facie rejection, MPEP § 2143.01 specifies that:

> Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either explicitly or implicitly in the references themselves or in the knowledge generally available to one of ordinary skill in the art.

In the present application, the various combinations of references proposed by the Examiner are not supported by a proper suggestion or motivation to make each proposed

modification. This means that the first criterion for a *prima facie* rejection has not been met, which in turn means the Examiner has failed to carry the burden of establishing a *prima facie* rejection. In addition, certain claim limitations are not taught or suggested by the cited combinations, which means that the third criterion for a *prima facie* rejection has not been met, and that the Examiner has further failed to carry the burden of establishing a *prima facie* rejection for this independent reason. Further, the Examiner has failed to put forth any arguments and has not provided any articulated reasoning as to how any deficiency (missing element) could be solved in a predictable manner through combination with any other reference.

B. Recent Decisions Affecting a Finding of Obviousness.

1. In re *Kahn*.

With respect to obviousness, a claimed invention is unpatentable if the differences between it and the prior art are "such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art."[59] Obviousness is a question of law, based upon underlying factual questions which are reviewed for clear error following a bench trial. These "underlying factual inquiries include: (1) The scope and content of the prior art; (2) The level of ordinary skill in the prior art; (3) The difference between the claimed invention and the prior art; and (4) Objective evidence of nonobviousness."[60]

In *Kahn* the Court noted that:

> ". . .to reject claims in an Application under § 103, an Examiner must show and unrebutted *prima facie* case of obviousness . . . on appeal to the board, an Applicant can overcome a rejection by showing insufficient evidence of a *prima facie* obviousness or by rebutting the *prima facie* case with evidence of secondary indicia of nonobviousness."[61] .

When combining references, it is well recognized that "[m]ost inventions arise from a combination of old elements and each element may often be found in the prior art."[62] "However,

---

[59] 35 U.S.C. § 103(a) (2000); *In re Kahn*, 441 F.3d 977, 985 (Fed. Cir. 2006) (citing *Graham v. John Deere Co.*, 383 U.S.1, 13-14, 86 S.Ct. 684, 15L, Ed. 2d 545, 1962)
[60] *In re Dembiczak*, 175 F.3d 994, 998 (Fed. Cir. 1999).
[61] *Kahn,* 441 F.3d at 985
[62] *In re Rouffett*, 149 F.3d 1350, 1357

mere identification in the prior art of each element is insufficient to defeat the patentability of the combined subject matter as a whole."[63] *Kahn* further states:

> Rather, to establish a *prima facie* case of obviousness based on a combination of elements disclosed in the prior art, the Board must articulate the basis on which it concludes that it would have been obvious to make the claimed invention. *Id.* In practice, this requires that the Board "explain the reasons one of the ordinary skill in the art would have been motivated to select the references and to combine them to render the claimed invention obvious." *Id. at 1357-59.* This entails consideration of both the "scope and content of the prior art" and the "level of ordinary skill in the pertinent art" aspects of the Graham test.[64]

The primary test that has been put forth by the Federal Circuit is the teaching-suggestion-motivation test. *Kahn* set forth that, when there is no explanation provided by the Board to explain the motivation, or the suggestion or the teaching, that would have led a skilled artisan at the time of the invention to the claimed combination as a whole, then the court would infer that hindsight was utilized to conclude that the invention was obvious. *Kahn* relied upon the *Rouffett* case for this teaching at 1358. The "teaching-suggestion-motivation" requirement was set forth to protect against the entry of hindsight into the obviousness analysis, a problem which §103 was meant to confront. Thus, in order to establish a *prima facie* case, some explanation as to teaching, suggestion, or motivation of each of the references and how they can be combined is required.

Although *Kahn* sets forth the teaching-suggestion-motivation test, there is still the "analogous-art" test that must be applied, this being one test that was articulated by the Supreme Court as part of the *Graham* analysis.[65] "The analogous-art test requires that the Board show a reference is either in the field of the Applicant's endeavor or is reasonably pertinent as to the problem with which the inventor was concerned in order to rely on that reference as a basis for rejection."[66] The following was further stated by *Kahn*:

> References are selected as being reasonably pertinent to the problem based on the judgment of a person having ordinary skill in

[63] *Kahn,* 441 F.3d at 986, citing *Rouffett*, 149 F.3d at 1355, 1357
[64] *Id.*
[65] See *Dann v. Johnston*, 425 U.S. at 219, 226, 96 S.Ct. 1393, 47 L.Ed 2d 692 (1976).
[66] *Kahn,* 441 F.3d at 987.

> the art. *Id.* ("It is necessary to consider the reality of the circumstances, in other words, common sense--in deciding in which fields a person of ordinary skill would reasonably be expected to look for a solution to the problem facing the inventor." (quoting *In re Wood*, 599 F.2d 1032, 1036 (C.C.P.A. 1979))). We have explained that this test begins the inquiry into whether a skilled artisan would have been motivated to combine references by defining the prior art relevant for the obviousness determination, and that it is meant to defend against hindsight. See *id.*; *In re Clay*, 996 F.2d 656, 659-60 (Fed. Cir. 1992).[67]

As such, the first step of analyzing the combination provided by the Examiner is to examine the references and determine if the combination satisfies the analogous-art test. The next step for determining obviousness is to analyze the teaching-suggestion-motivation test which:

> . . . picks up where the analogous art test leaves off and informs the Graham analysis. To reach a non-hindsight driven conclusion as to whether a person having ordinary skill in the art at the time of the invention would have viewed the subject matter as a whole to have been obvious in view of multiple references, the Board must provide some rationale, articulation, [**23] or reasoned basis to explain why the conclusion of obviousness is correct. The requirement of such an explanation is consistent with governing obviousness law, see *§ 103(a)*; *Graham*, 383 U.S. at 35; *Dann*, 425 U.S. at 227-29, and helps ensure predictable patentability determinations.[68]

Even if all of the elements of a claim are disclosed in various prior art references, the long-standing rule that a claimed invention, as a whole[69], cannot be said to be obvious unless there is some reason or motivation given in prior art why someone would have been prompted to combine the teachings or the references.[70] The prior art itself may suggest desirability of a combination, or the motivation may come from other sources (for example, economic factors).[71] Thus, the motivation to combine the relevant art or teachings does not have to be found explicitly in the prior art but, rather, can be implicit thereto. "However, rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated

---

[67] *Kahn,* 441 F.3d at 987
[68] *Id.*
[69] *In re Hiraro*, 535 F.2d, 67, (C.C.P.A. 1966).
[70] *In re Regel*, 526 F.2d, 1399 (C.C.P.A. 1975); *In re Bond*, 910 F.2d, 831, (Fed. Cir. 1990).
[71] See e.g. *In re Clinton*, 527 F.2d 1226 (C.C.P.A. 1976); *Cable Elec. Prods., Inc. v. Genmart, Inc.*, 77 F.2d, 1015 (Fed. Cir. 1985).

reasoning with some rational underpinning to support the legal conclusion of obviousness."[7273] The purpose of such requirement is to ensure "due process and non-arbitrary decision making", as it is in § 103.[74]

Kahn articulated the considerations for motivation when analyzing obviousness. The Court stated "the problem examined is not the specific problem solved by the invention, but the general problem that confronted the inventor before the invention was made."[75] In the reference in *Cross*, the quote that was cited by the Court[76] was that "one of ordinary skill in the art need not see the identical problem addressed in the prior art reference to be motivated to apply its teachings." As to motivation, the Courts upheld that the evidence of motivation to combine the prior art references "may flow from the prior art references themselves, knowledge of one of ordinary skill in the art, or, in some cases, from the nature of the problem to be solved."[77] *Kahn* summarized the motivation-suggestion-teaching test as follows:

> Therefore, the "motivation-suggestion-teaching" test asks not merely what the references disclose, but whether a person of ordinary skill in the art, possessed with the understandings and knowledge reflected in the prior art, and motivated by the general problem facing the inventor, would have been led to make the combination recited in the claims. See *Cross Med. Prods.*, 424 F.3d at 1321-24. From this it may be determined whether [**26] the overall disclosures, teachings, and suggestions of the prior art, and the level of skill in the art–i.e., the understandings and the knowledge of persons having ordinary skill in the art at the time of the invention–support the legal conclusions of obviousness. See *Princeton Biochemicals*, 411 F.3d at 1338 (pointing to evidence supplying detailed analysis of the prior art and the reasons one of ordinary skill would have possessed the knowledge and motivation to combine).[78]

---

[72] *Kahn*, 441 F.3d at 998 referring to *Lee*, 277, F.3d at 1343-46 and *Rouffett*, 149 F.3d at 1355-59.
[73] It is noted that the Supreme Court in the recently decided case, *KSR International Co. v. Teleflex Inc, et al.*, 127 S. Ct. 1727 (2007) cited this specific language at page 1741 therein.
[74] *Kahn*, 441 F.3d at 998 referring to *Lee*, 277, F.3d at 1343-46 and *Rouffett*, 149 F.3d at 1355-59.
[75] *Id.* at 988, referring to *Cross Medical Products, Inc. v. Metronics Sofamore Danek, Inc.*, 424 F.3d 1293, 1323 (Fed. Cir. 2005).
[76] *Cross*, 424 F.3d at 1323.
[77] *Medichem S.A. v. Rolabo, S.L.*, 437 F.3d 1157, 1165 (Fed. Cir 2006), quoting *Brown and Williamson Tobacco Corp. v. Phillip Morris, Inc.*, 229 F.3d, 1120, 1125 (Fed. Cir. 2000).
[78] *Kahn,* 441 F.3d at 988.

In *Alza Corporation v. Mylan Laboratories, Inc.*, 464 F.3d 1286 (Fed. Cir. 2006), the Federal Circuit has responded to arguments made during pendency of the recently decided Supreme Court case, *KSR International Co v. Teleflex Inc, et al.*, 127 S. Ct. 1727 (2007) and has spelled out its law on obviousness, insisting that it is in harmony with Supreme Court precedent.

In the facts of this case, *Alza* sued Mylan for infringement of its patent (6,124,355) under 35 U.S.C. §271(e)(2) after Mylan sought FDA approval to market a generic version of oxybutynin, a drug used to treat urinary incontinence. The Federal Circuit affirmed the obviousness and non-infringement decisions of the district court.

In the process, Judge Arthur Gajarsa dedicated five pages of his opinion to then outline the Federal Circuit's law on obviousness, responding to many arguments made in the then pending Supreme Court case of *KSR Int'l Co. v. Teleflex, Inc.* (U.S. No. 04-1350). KSR and many amici, including the U.S. government, have challenged the Federal Circuit rule that proof of obviousness must include a showing of a "teaching, suggestion, or motivation" to combine the prior art elements of the claimed invention. *KSR* and others have said that this requirement is too rigid and is inconsistent with Supreme Court decisions issued since *Graham v. John Deere Co.*, 383 U.S. 1 (1966).

Judge Gajarsa wrote the following in his *Alza* opinion:

> This requirement has been developed consistent with the Supreme Court's obviousness jurisprudence as expressed in *Graham* and the text of the obviousness statute that directs us to conduct the obviousness inquiry "at the time the invention was made" 35 U.S.C. §103. As we explained in [*In re Kahn*, 441 F.3d 977 (Fed. Cir. 2006)],
>
> > The motivation-suggestion-teaching test picks up where the analogous art test leaves off and informs the *Graham* analysis. To reach a non-hindsight driven conclusion as to whether a person having ordinary skill in the art at the time of the invention would have viewed the subject matter as a whole to have been obvious in view of multiple references, the Board must provide some rationale, articulation, or reasoned basis to explain why the conclusion of obviousness is correct. The requirement of such an explanation is consistent with governing obviousness law . . .

441 F.3d at 987. We further explained that the "motivation to combine" requirement "[e]ntails consideration of both the 'scope and content of the prior art' and 'level of ordinary skill in the pertinent art' aspects of the *Graham* test." *Id*. at 986.

At its core, our anti-hindsight jurisprudence is a test that rests on the unremarkable premise that legal determinations of obviousness, as with such determinations generally, should be based on evidence rather than on mere speculation or conjecture. Our court's analysis in *Kahn* bears repeating:

> A suggestion, teaching, or motivation to combine the relevant prior art teachings *does not have to be found explicitly in the prior art*, as "the teaching, motivation, or suggestion may be implicit from the prior art as a whole, rather than expressly stated in the references.... The test for an implicit showing is what the combined teachings, knowledge of one of ordinary skill in the art, and the nature of the problem to be solved as a whole would have suggested to those of ordinary skill in the art." However, rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be *some* articulated reasoning with *some* rational underpinning to support the legal conclusion of obviousness. This requirement is as much rooted in the Administrative Procedure Act [for our review of Board determinations], which ensures due process and non-arbitrary decision making, as it is in § 103.

441 F.3d at 987-88 (quoting *In re Kotzab*, 217 F.3d 1365, 1370 (Fed. Cir. 2000)) (citations omitted) (emphases added). There is flexibility in our obviousness jurisprudence because a motivation may be found *implicitly* in the prior art. We do not have a rigid test that requires an actual teaching to combine before concluding that one of ordinary skill in the art would know to combine references. This approach, moreover, does not exist merely in theory but in practice, as well. Our recent decisions in *Kahn* and in [*Cross Med. Prods., Inc., v. Medtronic Sofamor Danek, Inc.*, 424 F.3d 1293 (Fed. Cir. 2005)] amply illustrate the current state of this court's views.[79]

2. KSR

The recently issued Supreme Court Case in *KSR* held that the Federal Circuit's Teaching, Suggestion or Motivation (TSM) test to combine known elements in order to show that the combination is obvious is too rigid. The Court reinforced their position that analysis under *Graham* has been reaffirmed. The Court indicated that its holding was that a "patent for a combination which only unites old elements with no change in their respective functions . . .

[79] *Alza Corporation v. Mylan Laboratories, Inc.*, 464 F.3d 1286, 1290 (Fed. Cir. 2006).

obviously withdraws what is already known into the field of its monopoly and diminishes the resources available to skillful men."[80] The Court stated that this was a "principal reason for declining to allow patents for what is obvious. The combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results."[81] The Court further went on to indicate that there were three cases that illustrated the application of this doctrine of predictability. The first case was *United States v. Adams*, 383 U.S. 39, 40 (1966). In discussing this case, the Court noted that it had "relied upon the corollary principal that when the prior art teaches away from combining certain known elements, discovery of a successful means of combining them is more likely to be non-obvious."[82] In the second case, *Anderson's-Black Rock, Inc. v. Pavement Salvage Co.*, 396 U.S. 57 (1969), the Court reiterated "while the combination of old elements performed a useful function, it added nothing to the nature and quality of the radiant-heat burner already patented."[83] In the third case, *Sakraida v. AGPro, Inc.*, 425 U.S. 273 (1976), the Court stated that "when a patent 'simply arranges old elements with each performing the same function it had been known to perform' and yields no more than one would expect from such an arrangement, the combination is obvious."[84]

The Court summarized these three cases as follows:

> The principles underlying these cases are instructive when the question is whether a patent claiming the combination of elements of prior art is obvious. When a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. *If a person of ordinary skill can implement a predictable variation, §103 likely bars its patentability.* For the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill. *Sakraida* and *Anderson's-Black Rock* are illustrative-a court must ask whether the improvement is more than the predictable use of prior

[80] *KSR,* 127 S. Ct. 1727, 1739 (2007), Citing *Great Atlantic & Pacific Co. v. Supermarket Equipment Corp.*, 340 U.S. 147, 152 (1950).
[81] *Id.*
[82] *Id.* at page 1740.
[83] *Id.*
[84] *Id.*, Citing *Sakradia* at 282.

art elements according to their established functions.[85] (Emphasis added.)[86]

The Court recognized that following the above stated principals might involve more than "the simple substitution of one known element for another or the mere application of a known technique to a piece of prior art ready for the improvement."[87] The Court noted that it might "be necessary for a Court to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent that issued."[88] However, the Court also noted that the analysis should be "made explicit" citing *Kahn* wherein it stated that "rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead there must be some articulated reason with some rational underpinning to support the legal conclusion of obviousness."[89] The Court noted that, however, "the analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ."[90]

Although the Court in this opinion rejected the rigidity of the TSM test, there was some reference to the decision in *Alza* wherein the Court noted the Federal Circuit's position that "there is flexibility in our obviousness jurisprudence because the motivation may be found *implicitly* in the prior art. We do not have a rigid test that requires an actual teaching to combine . . . ," citing *Alza*, 464 F.3d at 1291.[91] However, the Court also noted that the *Alza* decision was not before it and that, although they may describe an analysis more consistent with the Court's earlier precedence, the Court of Appeals would have to consider the current decision in view of its future cases.

---

[85] *KSR*, 127 S. Ct. at page 1741.
[86] *Id.*
[87] *Id.*
[88] *Id.* at page 1741
[89] *Id.*
[90] *Id.*.
[91] *Id.* at page 1743.

C. 35 U.S.C § 103 Rejection in the Application on Appeal.

Regarding Claims 1-2, 5, 10-11, 15-17, 20, 25-26 and 30 , the Examiner states in the Final Office Action dated September 8, 2006:

> Tarr teaches the invention substantially as claimed (As in claim 16) including a system for distributing software, comprising: a television broadcast distribution system having one or more broadcast channels for broadcasting analog and digital television information to a receiver of a user (col. 6, lines 44-65); one or more discrete software data streams designated for transmission on select ones of said one or more broadcast channels, each of the one or more discrete data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique ID'S *(sic)* is unique to a user (col. 8, lines 46-57, the streams can be encrypted, therefore they would have a key unique to the user); a user storage device connected to said receiver through a monitoring interface and said user storage device operable to store both one or more of the unique ID'S *(sic)* associated with the user prior to downloading of the one or more discrete software data streams and, after downloading thereof, for storing said selected one or more discrete software data streams wherein each of said unique ID'S *(sic)* stored in said user interface is stored there by the user (col. 8, lines 46-57, the key to decrypt the software would be stored on the user's machine); wherein said one or more software data streams are transmitted over said selected ones of said one or more broadcast channels at a scheduled time, each of said one or more discrete software data streams having associated therewith a unique ID associated therewith (col. 6, lines 44-65); wherein said select ones of said one or more discrete software data streams are downloaded via said receiver to said monitoring interface for filtering said discrete software data streams according to said respective unique ID'S *(sic)* (col. 8, lines 46-57); however, Tarr does not explicitly teach a system where after downloading of one or more unique data streams and wherein the unique ID is deleted after downloading.
>
> Birdwell teaches associating a user with a user storage device associated with the user operable to store both one or more of the unique ID'S *(sic)* associated with the user prior to downloading of one or more discrete software data streams and after downloading of one or more unique data streams and wherein the unique ID is deleted after downloading (col. 10, lines 57-67 and col. 1, lines 1-25).
>
> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Tarr regarding software download to a set-top box with the teachings of Birdwell regarding deleting ID'S *(sic)* after downloading is complete because deleting already used ideas would save space in memory 4. As to

claim 1 7, Tarr teaches a system wherein a cable television broadcast system is used (col. 6, lines 44-65).[92]

Regarding Claims 3, 4, 18 and 19, the Examiner states in the Final Office Action dated September 8, 2006:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell regarding a system for downloading software with the teachings of Metz regarding checking for errors because a downloaded data stream would be useless if it had errors.[93]

Regarding Claims 6, 12-14, 21 and 27-29, the Examiner states in the Final Office Action dated September 8, 2006:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell combination regarding a system for downloading software with the teachings of Brandenburg regarding broadcasting a data stream once because broadcasting a data stream only once reduces the bandwidth necessary for software transmission (Brandenburg, col. 1, lines 36-51).[94]
>
> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Tarr-Birdwell regarding a system for downloading software with the teachings of Brandenburg for distributing software from a receiver to a user computer because downloading software to a receiver device before installing it on a user computer enhances security (col. 4, lines 27-61).[95]
>
> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell-Brandenburg combination regarding to *(sic)* computers linked together with the idea of using a universal serial bus to link the computers because a universal serial bus is an easy and efficient way to link to computers together.[96]

---

[92] See Final Office action mailed September 06, 2006, page 3 paragraph 3.
[93] See Final Office action mailed September 06, 2006, page 4 paragraph 11.
[94] See Final Office action mailed September 06, 2006, page 5 paragraph 15.
[95] See Final Office action mailed September 06, 2006, page 6 paragraph 16.
[96] See Final Office action mailed September 06, 2006, page 6 paragraph 17.

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell-Brandenburg combination regarding to *(sic)* computers linked together with the idea of using a high-performance serial bus is an easy and efficient way to link to computers together.[97]

Regarding Claims 7-8 and 22-23, the Examiner states in the Final Office Action dated September 8, 2006:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell combination regarding the distribution of software with the teachings of Durden regarding the storage and uploading of accounting data because an accounting device is useful for billing for broadcast services (Durden, col. 2, lines 11-39).[98]

Regarding Claims 9 and 24, the Examiner states in the Final Office Action dated September 8, 2006:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell-Durden combination regarding the distribution of software and the uploading of accounting data with the teachings of Fries regarding a television set-top box connected to the internet because transmitting data over a packet-switched global network would be a more modern alternative to Durden method of sending accounting data over the telephone network.[99]

Appellants submit that the Examiner simply has broken Appellants' invention into its component parts and then attempted to find a prior art reference corresponding to each component to support an obviousness rejection under 35 U.S.C. § 103. In order to establish a *prima facie* case of obviousness using the combinations of *Tarr* and *Birdwell*; *Tarr*, *Birdwell* and *Brandenburg*; *Tarr*, *Birdwell* and *Durden*; and *Tarr*, *Birdwell*, *Durden* and *Fries*, the Examiner must show the teachings therein and the suggestions associated therewith, in addition to the level

[97] See Final Office action mailed September 06, 2006, page 7 paragraph 18.
[98] See Final Office action mailed September 06, 2006, pages 7 and 8 paragraph 21.
[99] See Final Office action mailed September 06, 2006, pages 8 and 9 paragraph 25.

of skill in the art and a discussion of why such person could implement a predictable variation, in order to support a conclusion of obviousness as it relates to the entire invention. Appellants submit that the Examiner's combinations of *Tarr*, *Birdwell*, *Brandenburg*; *Durden* and *Fries* are conclusory, and that no articulated reasoning with some rational underpinning to support the combinations has been provided nor has the Examiner made any arguments as to how such combination(s) would constitute a predictable variation by one skilled in the art. Further, Appellants submit that support for the combinations is based on hindsight and that the combinations are improper.

1. **Independent Claims 1 and 16 as rejected by the combination of *Tarr* and *Birdwell*.**

In the Final Office Action mailed September 06, 2006, the Examiner maintains his 35 U.S.C. § 103 rejection of Claims 1-30. On page 2, paragraph 3 of the Final Office Action, the Examiner states:

> "Tarr teaches the invention substantially as claimed (as in claim 16) including a system for distributing software, comprising: a television broadcast distribution system having one or more broadcast channels for broadcasting analog and digital television information to a receiver of a user (col. 6, lines 44-65); one or more discrete software data streams designated for transmission on select ones of said one or more broadcast channels, each of the one or more discrete data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique ID's *(sic)* is unique to a user (col. 8, lines 46-57, the streams can be encrypted, therefore they would have a key unique to the user); a user storage device connected to said receiver through a monitoring interface and said user storage device operable to store both one or more of the unique ID's *(sic)* associated with the user prior to downloading of the one or more discrete software data streams and, after downloading thereof, for storing said selected one or more discrete software data streams wherein each of said unique ID's *(sic)* stored in said user interface is stored there by the user (col. 8, lines 46-57, the key to decrypt the software would be stored on the user's machine); wherein said one or more software data streams are transmitted over said selected ones of said one or more broadcast channels at a scheduled time, each of said one or more discrete software data streams having associated therewith a unique ID associated

> therewith (col. 6, lines 44-65); wherein said select ones of said one or more discrete software data streams are downloaded via said receiver to said monitoring interface for filtering said discrete software data streams according to said respective unique ID's *(sic)* (col. 8, lines 46-57);. . ." [100]

In the same paragraph, the Examiner further states "however, Tarr does not explicitly teach a system where after downloading of one or more unique data streams and wherein the unique ID is deleted after downloading."[101] The Examiner offers to combine *Tarr* with the *Birdwell* reference to "[teach] associating a user with a user storage device associate with the user operable to store both one or more of the unique ID's *(sic)* associated with the user prior to downloading of one or more discrete software data streams and after downloading of one or more unique data streams and wherein the unique ID is deleted after downloading (col. 10, lines 57-67; col. 11, lines 1-25)."[102] The Examiner concludes that "It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Tarr regarding software download to a set-top box with the teachings of Birdwell regarding deleting ID's *(sic)* after downloading is complete because deleting already used ideas would save space in memory."[103]

2. The Cited References – Teaching/Suggestion/Motivation Test.

The primary step in determining obviousness is to analyze under the teaching-suggestion-motivation test. As previously discussed, the recent *KSR* Supreme Court case indicated that the Teaching-Suggestion-Motivation (TSM) test is not a rigid test; however, it is still considered to be a factor. Under this test, each of the references must contain some type of teaching, as well as some type of suggestion, to allow for the combination. One also must be motivated to combine the references. If this test alone were utilized, the questions that must be answered are whether *Tarr* and *Birdwell* contain any teaching that would suggest to one skilled in the art to combine these two references to overcome the problem addressed by the present application, and whether any motivation to so combine exists.

---

[100] See Final Office action mailed September 06, 2006, pages 2 and 3 paragraph 3.
[101] See Final Office action mailed September 06, 2006, page 3 paragraph 3.
[102] See Final Office action mailed September 06, page 3 paragraph 3.
[103] See Final Office action mailed September 06, 2006, page 3 paragraph 3.

a. Discussion of U.S. Patent No. 6,238,290 to *Tarr et al*

The primary reference cited by the Examiner is *Tarr*. The primary purpose of *Tarr* is to allow a game manufacturer to distribute software associated with their games on a broadcast channel at pre-selected times.[104] In the preferred embodiment, this provides for distribution of the software in 30 minute time slots, such that two different games will be distributed in the course of an hour or there may be multiple games during each half-hour time slot that could be selected by the user.[105] A subscriber ("user") must obtain a start-up CD-ROM.[106] The CD-ROM contains the executable code and art for the video game platform chosen by the subscriber.[107] The CD-ROM also contains the device drivers and software which enables a primary processor to select and receive software programs from a head-end server.[108] The CD-ROM further contains a program to decrypt the broadcast programs.[109] The CD-ROM controls the operation and provides information to the primary processor that is utilized for actually decrypting encrypted programs.[110]

The system comprises the primary processor, a secondary processor and a head-end server.[111] The primary processor is operable to interface with the broadcast channel, download information and software, decompress and decrypt the received information for storage in a local hard disk.[112] The secondary processor is operable to interface with the CD-ROM and initiate the system.[113] A monitor is connected to the secondary processor.[114]

The monitor displays a logo screen upon startup. However, the user interface screen may appear on the monitor even though the primary processor is not turned on.[115] The head-end server broadcasts, to the primary processor, the time-slots and available programs periodically.[116]

---

[104] See *Tarr* Abstract, Col 8, lines 30-35.
[105] See *Tarr* Col. 7, lines 40-43.
[106] See *Tarr* Col. 5, line 67 – Col. 6, line 2.
[107] See *Tarr* Col. 5, line 59 – Col. 6, line 6.
[108] See *Tarr* Col. 5, lines 45-51.
[109] See *Tarr* Col. 5, lines 51-54.
[110] See *Tarr* Col. 5, line 45 – Col. 6, line 6.
[111] See *Tarr* Col. 4, line51 – Col. 5, line 15.
[112] See *Tarr* Col. 5, lines 9-24, Col. 6, lines 7-15.
[113] See *Tarr* Fig. 1, Col. 5, lines 12-15 and lines 45-46.
[114] See *Tarr* Col. 5, lines 43-44.
[115] See *Tarr* Col. 7, lines 10-16.
[116] See *Tarr* Col. 6, lines 23-30.

The secondary processor converts the broadcast data for display on the monitor.[117] The monitor displays game listings available to the subscriber.[118] By the user selecting one of these programs, the secondary processor instructs the main processor to download and temporarily store this program on the hard disk.[119] The user, while waiting for the game to download, can play other games stored on the CD-ROM.[120]

Further, the user may purchase a game.[121] If the user chooses to purchase a game, the monitor displays the time for delivery and purchase price.[122] The order is processed and the item is shipped.[123] This indicates that the user receives some type of CD-ROM or media with the game from another location and that this is not downloaded from the broadcast channel. Thus, if the user desires to have a copy of the game on their system for playing, other than temporarily, then the program must be purchased and received on a different media independent of the broadcast channel.

Independent Claim 1 of the instant application, as currently presented, is directed, in the preamble, to a method for distributing software. Independent Claim 16 of the instant application, as currently presented, is directed, in the preamble, to a system for distributing software. The Examiner has used independent Claim 16 to illustrate the 35 U.S.C. § 103(a) rejection. Therefore, Appellants will address this in kind. However, the deficiencies noted with respect to independent Claim 16 apply equally to the method claimed by independent Claim 1.

The first element of Claim 16 of the instant application is to provide a television broadcast distribution system having one or more broadcast channels for broadcasting analog and digital television information to a receiver of a user. The second element requires one or more discrete software data streams designated for transmission on select ones of said one or more broadcast channels, each of the one or more discrete data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user. The

---

[117] See *Tarr* Col. 7, lines 3-5.
[118] See *Tarr* Col. 7, lines 30-40.
[119] See *Tarr* Col. 6, lines 44-65.
[120] See *Tarr* Col. 7, line 64 – Col. 8, line3.
[121] See *Tarr* Col. 8, lines 30-45.
[122] See *Tarr* Col. 8, lines 32-35.
[123] See *Tarr* Col. 8, lines 39-43.

Examiner maintains the 35 U.S.C. § 103(a) rejection on the basis that *Tarr* discloses the use of a unique ID. The Examiner cites to the disclosure in *Tarr* at Column 8, lines 46-57. The Examiner states, in support of the obviousness rejection, that "since the streams can be encrypted, they therefore would have a key unique to the user."[124] The portion of the specification at issue is as follows:

> The advantages of the present invention are numerous. The present invention supports cost effective and convenient (for the subscriber and the network provider) distribution of large amounts of data. The present invention also uses limited bandwidth on analog cable thereby eliminating expensive cable network upgrades. Furthermore, the present invention allows the network provider to deliver software written for video game and personal computer systems with minimal changes to the software/title as used in a stand-alone environment, i.e., addition of a wrapper, *encryption*, etc. Still further, the present invention is compatible with various video game systems available for purchase.[125] (*emphasis added*).

However, as Appellants previously stated, the encryption is not unique to the user.[126] *Tarr* requires that each subscriber or user obtain a CD-ROM. The CD-ROM is a start-up CD-ROM. The specific disclosure sets forth as follows:

> To start up the system, the user inserts a *start-up CD-ROM* 38 into the CD-ROM drive 34. The CD-ROM 38 contains the device drivers and software which enable the PC or video game player to interact with the primary processor 18. The CD-ROM 38 also contains the software which enables the primary processor 18 to select and receive software programs from the head-end server 12. *To prevent unauthorized acquisition of the video games and other content, the CD-ROM 38 also contains a program to decrypt the broadcast programs.* The CD-ROM 38 may also contain multi-media content (e.g., graphics, animations, video clips and audio clips), software tools, programs (e.g., navigator), and software engines which can be used alone or with data that is broadcast over the network.[127] (Emphasis added).

Tarr clearly sets forth that "the CD-ROM also contains a program to decrypt the broadcast programs." Each user is required to obtain a CD-ROM corresponding to the platform

[124] See Final Office Action mailed September 06, 2006, page 2, paragraph 3.
[125] See *Tarr* Col. 8, lines 46-57.
[126] See Response dated August 07, 2006, page 9.
[127] See *Tarr* Col. 5, lines 45-58.

available in his/her home. In its Response, Appellants stated that nothing indicates that the encryption/decryption algorithm is unique to the user.[128] Rather, the encryption/decryption algorithm could be unique only to the start-up CD-ROM. However, it is very likely that every CD-ROM would contain the same encryption/decryption algorithm. Additionally, *Tarr* does not disclose a requirement that the encryption/decryption algorithm be unique to each CD-ROM. *Tarr* only requires that the user obtain a CD-ROM corresponding to the game platform in his/her home. This language clearly illustrates this is to ensure that any subscriber obtaining a start-up CD-ROM will be allowed to access the information. Thus, individuals not having access to the CD-ROM do not posses the ability to access the downloaded data from the broadcast channels. Furthermore, the decryption algorithm would be the same for all data streams downloaded, regardless of the vendor and regardless of the user. Therefore, there is no suggestion or teaching that would in any way associate this decryption operation with the particular user and/or vendor. As such, there is no unique ID that is associated with a software data stream of a software vendor or which unique ID is unique to a user. Thus, *Tarr* Col. 8, lines 46-57 does not support the use of the unique ID limitation in the claims.

Additionally, the claims require that the unique ID be associated therewith by an associated software vendor. Not only does *Tarr* not disclose that, even if it were considered a unique ID which Appellants submit is not, the encryption/decryption algorithm would be associated to each software data stream and unique to each user; *Tarr* further does not disclose that the associating of this encryption/decryption algorithm to each software data stream would be done by an associated software vendor. *Tarr's* system would have the same algorithm regardless of the software vendor.

The Examiner, in response to Appellants' arguments filed August 8, 2006, stated that the claims of the instant application require that the software data streams have a unique ID associated therewith and that the unique ID is unique from a user's perspective. The Examiner states:

> Applicant's arguments filed 8/7/2006 have been fully considered but they are not persuasive. The applicant's arguments would be valid if the claim language does not limit the applicant's invention to this scenario. Instead, the claim language indicates that the software data streams have a unique ID associated therewith and that the unique ID is unique to a user so in other words the unique ID could uniquely identify the software stream and the unique ID identifying the software stream is unique to the

[128] See Response dated August 07, 2006, page 10.

> user, according to the user's perspective. Therefore, the system taught by Tarr reads on the invention as claimed because the keys for the encrypted software uniquely identify the software stream and the users with the keys for the encrypted software are unique ID's for the software to the user. The fact that Tarr teaches that multiple users can have the same key is irrelevant in Tarr, because that key still is a unique ID to the user for the encrypted software.[129]

The Examiner and Appellants agree that the unique ID renders the claims of the instant application allowable. The claim requires one or more discrete software data streams designated for transmission on select ones of the one or more broadcast channels, each of the one or more discrete data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of *the unique IDs is unique to a user*. Appellants and the Examiner disagree on the meaning of "unique to a user." The Examiner's interpretation of the instant claim still does not render the claim obvious in light of *Tarr*. To use the Examiner's interpretation would result in multiple software data streams with the same ID, regardless of vendor, which hardly is unique from a user's perspective. However, when reading the instant claim in light of the specification, the unique ID is provided by a software vendor to a user to uniquely identify the user.[130] Those skilled in the art would understand what is claimed when the claim is read in light of the specification.[131]

The third element of Independent Claim 16 requires a user storage device associated with the user and connected to the receiver through a monitoring interface and the user storage device operable to store both one or more of the unique IDs associated with the user prior to downloading of the one or more discrete software data streams *wherein each of the unique IDs stored in the user interface is stored there by the user*. The Examiner again cites *Tarr*, Col 8, lines 46-57, stating "the key to decrypt the software would be stored on the user's machine."[132] However, *Tarr* does not disclose that the decryption algorithm exists any place other than on the CD-ROM.[133] The CD-ROM only enables the primary processor to select and receive

---

[129] See Final Office Action mailed September 06, 2006, page 9, paragraph 27.
[130] See Specification page 41, line 24 – page 42, line 25.
[131] *Orthokinetics, Inc., v. Safety Travel Chairs, Inc.*, 806 F.2d 1565, 1576 (Fed. Cir. 1986).
[132] See Final Office Action mailed September 06, 2006, page 3, paragraph 3.
[133] See *Tarr* Col 5, lines 51-54.

software.[134] The secondary processor generates operating instructions for the primary processor.[135] The user does not store the decryption algorithm anywhere. The CD-ROM enables download of software. There would be no motivation to have the user first relocate the encryption/decryption algorithm to the user interface, then still use the CD-ROM to initialize the system and direct the primary processor regarding the download operation. As such, there is no suggestion or teaching for storing in the selected one or more discrete software data streams wherein each of the unique IDs stored in the user interface is stored there by the user.

The next element of Independent Claim 16 requires that the one or more discrete software data streams are transmitted over the select ones of the one or more broadcast channels at a scheduled time, and that *each of the one or more discrete software data streams having associated therewith the associated one of the unique IDs*. The Examiner indicated that this element of the claims is obviated by the language of *Tarr* at Col. 6, lines 44-65. This language in *Tarr* is set out as follows:

> Turning now to FIG. 4, the method of the present invention may be more particularly described. As indicated above, the method is specifically directed for delivering a software program to a subscriber over a cable network which includes a head-end server. In operation, a plurality of software programs are compressed 39 at the head-end server and transmitted 41 over the cable network at corresponding predetermined scheduled times. A primary processor comprising, for example, a customized low end PC and hard disk is provided 43 in communication with the cable network. A secondary processor comprising, for example, a game machine which includes a game CPU and CD-ROM drive is provided 45 in communication with the primary processor. Operating instructions are generated 47 at the secondary processor for receipt by the primary processor so as to enable the primary processor to select and receive one of the plurality of software programs from the cable network without upstream communication to the head-end server. The selected software program is thereafter received 49 at the primary processor at the corresponding predetermined scheduled time whereupon it is decompressed 51 and temporarily stored 53.[136]

[134] See *Tarr* Col 5, lines 49-51.
[135] See *Tarr* Col 6, lines 57-60.
[136] See *Tarr* Col. 6, lines 44-65.

In its Response, Appellants commented that the only indication provided with respect to any unique ID is that associated with the decompression on the fly operation.[137] This portion of the specification does not require any decryption *per se*. The only support for decryption is found in the specification at Col. 5, lines 52-54. The decompression operation is not disclosed as a function of an encryption/decryption algorithm. Therefore, this portion of the specification does not suggest or teach a unique ID.

The Examiner also indicated that *Tarr* renders obvious the language of the claim "wherein said selected ones of said one or more discrete software data streams are downloaded via said receiver to said monitoring interface for filtering said discrete software data streams according to said respective unique IDs." Specifically, the Examiner cites *Tarr* at Col. 8, lines 46-57. However, no disclosure in *Tarr* provides for any "filtering." The encryption/decryption algorithm is merely a way to *allow the system* to open encrypted software.[138] The software is not downloaded in accordance with any respective unique ID. *Tarr* provides that a user selects software from a list for download. The software is downloaded and the user has the option of playing a stored game while waiting for the download. A unique ID to filter the data stream is not required as the only software that is downloaded is the software selected by the user. *Tarr* does not suggest or teach the use of one or more data streams downloaded via a receiver to a monitoring interface for *filtering said discrete software data streams according to said respective unique IDs*.

The final element of Independent Claim 16 requires that the unique ID for each of the one or more discrete software data streams downloaded is deleted from the user storage device after downloading thereof. The Examiner and Appellants agree that *Tarr* does not teach the aspect of downloading one or more unique data streams wherein the unique ID is deleted after downloading.

As such, the only disclosure in *Tarr* is to provide a system to download games from a server while using a start-up CD-ROM. There is no teaching or suggestion in *Tarr* that would provide for a unique ID associated with one or more discrete data streams, which unique ID for each of the one or more discrete software streams is associated therewith by an associated

[137] See Response dated August 07, 2006, page 11.
[138] See Response dated August 07, 2006, page 11.

software vendor and each of the unique IDs is unique to a user. A unique ID, uniquely identifying the user, in the *Tarr* system does not enhance or further the system for broadcasting updated or new software for varying gaming platforms. Basically, *Tarr* is concerned with providing gaming software of varying platforms to users who obtain a start-up CD-ROM for their respective platform. Any user with the start-up CD-ROM can select the software for download. However, in Appellants' present invention, only a specific user may access the software for download. Another user, using the same unique ID, could not access the software. *Tarr* does not contain a motivation that would cause one to search further and look for a solution of somehow providing a unique ID associated with a software data stream that is unique to the user for the purpose of filtering the broadcast software data stream based on the unique ID, wherein the unique ID is deleted after the software data stream is downloaded. There is no reason for such in order to achieve the purpose of *Tarr* and, therefore, there is no reason for one, faced with the problems to be solved by *Tarr*, to seek a solution to provide a unique ID that is associated with the software data stream and unique to the user such that only the user would have access to the software and, subsequent to the download, the unique ID is deleted.

The disclosure in *Tarr* does not support the Examiner's argument that a decryption algorithm stored on a start-up CD-ROM would be the same as a unique ID that is unique to a user.[139] *Tarr* does not provide support for a unique ID associated by an associated vendor. Additionally, *Tarr* contains no support for the user storing the unique ID in the user interface.[140] Furthermore, *Tarr* contains no support for filtering the software stream based on the unique ID that is unique to the user.[141] *Tarr* discloses a start-up CD-ROM used to instruct a primary processor how to download software. Inclusion of the encryption/decryption algorithm as part of the start-up CD-ROM provides a convenient way for users to download software compatible with their gaming platforms and maintain memory space in their computers (primary processors).[142] One skilled in the art would not look to any other reference to provide a convenient way to save memory, as that is what *Tarr* already provides.

---

[139] See Final Office Action dated September 06, 2006, page 2, paragraph 3.
[140] See Final Office Action dated September 06, 2006, page 2, paragraph 3.
[141] See Final Office Action dated September 06, 2006, page 3, paragraph 3.
[142] See *Tarr*, Col. 5, lines 45-58.

Thus, to apply *Tarr* for the purpose of obviating Claim 1 in the instant application, the Examiner must show that *Tarr* contains a teaching, suggestion, or motivation to solve the problem faced by Appellants' present claims. *Tarr* must suggest that, at the time of the invention, a problem existed that could be solved by incorporating a unique ID that is unique to the user in the *Tarr*-system for the purpose of filtering software data streams associated with the unique ID, then deleting the unique ID from the user interface after download of the software data streams. *Tarr* does not contain any such teaching, suggestion or motivation.

b. Discussion of U.S. Patent No. 6,002,852 to *Birdwell et al.*

The Examiner has provided *Birdwell* to cure the deficiencies in *Tarr*. Specifically, the Examiner has relied on *Birdwell*, "[to teach] associating a user with a user storage device associated with the user operable to store both one or more of the unique ID's *(sic)* associated with the user prior to downloading of one or more discrete software data streams and after downloading of one or more unique data streams and wherein the unique ID is deleted after downloading (col. 10, lines 57-67 and col. 11, lines 1-25)."[143] The Examiner further states:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Tarr regarding software download to a set-top box with the teachings of Birdwell regarding deleting ID's *(sic)* after downloading is complete because deleting already used ideas would save space in memory."[144]

The primary purpose of *Birdwell* is to provide a method for downloading data from a server to a client.[145] Initially, the server receives the data from a provider of the data. The server stores the data for downloading to specific clients.[146] The provided data can be designated for multiple clients.[147] The *Birdwell* system uses two methods to download data.[148] The first download method is a point-to-point connection.[149] In the point-to-point connection, the client has a defined address on the network. The server is aware of the client's defined address. The

[143] See Final Office action mailed September 06, 2006, page 3 paragraph 3.
[144] See Final Office action mailed September 06, 2006, page 3 paragraph 3.
[145] See *Birdwell* Abstract.
[146] See *Birdwell* Col. 3, lines 36-40.
[147] See *Birdwell* Col. 3, lines 14-57.
[148] See *Birdwell* Col. 3, lines 3-29.
[149] See *Birdwell* Col. 3, lines 7-9.

server can connect directly to the client and download the data using this defined address.[150] The second method of delivery is a "broadcast mode."[151] The server determines that there are a plurality of clients that need to receive the data.[152] The server calculates the most efficient method to download the data.[153] If the server determines it is more efficient to use the broadcast method, the server can broadcast the data to a specific address that the clients have access to.[154] This information will be transmitted to the particular clients such that they will then be set up to receive on that address.[155] The server provides all clients in the system with an address on a global network such that information can be sent thereto as to what address to then change to in order to receive downloaded data.[156] Thereafter, the server broadcasts the data to that address and each of the clients will receive it.[157] The server assigns a download identification for each broadcast.[158] The clients who receive the broadcast use this download identification to confirm receipt of the broadcast.[159]

The *Birdwell* system is divided into two application kernels. The first is the one for downloading the data and the second is the one for confirming the download. The purpose of *Birdwell* is to ensure data is sent. This is apparent in that *Birdwell* allows another entity to deposit data on the server and then the server takes care of delivering that data. *Birdwell's* only concern is to confirm that the data was sent. *Birdwell* provides a method for the efficient and opportunistic downloading of software data.

Appellants submit that the addition of *Birdwell* does not render Appellants' invention obvious. *Birdwell* does not cure the deficiencies of *Tarr*. As stated hereinabove, the Examiner is using *Birdwell* to teach the deleting of a unique ID after the downloading of one or more unique data streams. The specific portion of the *Birdwell* disclosure cited by the Examiner states:

> FIG. 13is a flow diagram of the Confirm Download routine of the server APP. In step 1301, the routine removes the client

[150] See *Birdwell* Col. 3, lines 9-14.
[151] See *Birdwell* Col. 3, lines 18-21.
[152] See *Birdwell* Col. 3, lines 23-29, lines 39-44, Col. 5, lines 12-14.
[153] See *Birdwell* Col. 3, lines 23-29, lines 44-65
[154] See *Birdwell* Col. 3, lines 44-47, Col. 5, lines 34-36.
[155] See *Birdwell* Col. 5, lines 36-39.
[156] See *Birdwell* Col. 6, lines 3-9, lines 34-36.
[157] See *Birdwell* Col. 6, lines 36-39.
[158] See *Birdwell* Col. 6, lines 39-40.
[159] See *Birdwell* Col. 6, lines 41-43.

> identification from the Pending Download table for the download identification for which the client confirmed receipt and returns.
>
> FIG. 14 is a flow diagram of the Request Download routine of the server FTC. The Request Download routine determines which transmission path to use and transmits the download data. Alternatively, this routine could be passed an indication as to whether the data should be sent by the point-to-point or should be broadcasted. In step 1401, the Request Download routine determines the size of the data to be downloaded. In step 1402, the Request Download routine invokes a routine provided by the server BDC to determine the transmission characteristics for the broadcast. In step 1403, the Request Download routine invokes a routine provided by the server point-to-point component to determine the transmission characteristics of the point-to-point transmission. In step 1404, the Request Download routine evaluates the transmission characteristics to select either the broadcast transmission mechanism or point-to-point connection. In step 1405, if broadcast mechanism is selected, then the routine continues at step 1406, else the routine continues at step 1407. In step 1406, the Request Download routine opens a broadcast stream. In step 1407, the Request Download routine opens a point-to-point stream. In step 1408, the Request Download routine writes the download data to the open stream and returns.[160]

Birdwell does not disclose a unique ID in accordance with the claims of the instant Application. Appellants' present inventive concept, as defined by the currently presented claims, requires a unique ID, wherein one or more discrete software data streams having the unique ID associated therewith, which unique ID for each of the one or more discrete data streams is *associated therewith by an associated software vendor* and each of the unique IDs is unique to a user. *Birdwell* discloses a unique ID for a client computer[161] and a download identification for each of the software streams. These are the IDs referenced by the portion of *Birdwell* cited by the Examiner. *Birdwell* uses the unique ID and download identification to confirm that the download was received. In its Response, Appellants stated *Birdwell* does not disclose that this ID was associated therewith by the software vendor.[162] *Birdwell* also does not disclose that the unique IDs are unique to a user and uniquely associated therewith. The specific portion of the *Birdwell* disclosure sets forth as follows:

---

[160] See *Birdwell* Col. 10, lines 57-67, Col 11, lines 1-25.
[161] See *Birdwell* Col. 5, lines 11-19.
[162] See Response dated July 14, 2005, page 11.

> Each *client computer system* is assigned a unique client identification number.[163] *(emphasis added)*
>
> *The server assigns a download identification to the broadcast*, which is transmitted with the broadcast. The clients who receive the broadcast use this download identification to confirm receipt of the downloaded data. At the time of the broadcast, clients A and B were connected to the server, but client C was not connected. The computer system of client C, however, was running and thus able to receive the broadcast. The computer system of client D was not running and thus was unable to receive the broadcast. As shown in step 202, once client A receives the broadcast, it confirms receipt of the download by sending a "confirm download" message to the server. Upon receiving the "confirm download" message from client A, the server removes client A's identification from the list of clients who have not yet confirm the receipt of the download data.[164] *(emphasis added)*.

Furthermore, the claims of the instant application require that the user associate, with a monitoring interface of the user connected to the receiver, this unique ID with the software data stream. *Birdwell* does not disclose any suggestion or teaching that the user utilizes a monitoring interface on the user system to associate the downloaded data with this download ID.

The last element, that of deleting the unique ID from the monitoring interface, is not present in *Birdwell*, nor is it suggested or even related to the purpose of *Birdwell*. *Birdwell* does not teach or suggest a unique ID that is stored at the monitoring interface. As such, *Birdwell* also does not teach or suggest that any unique ID would be deleted from the monitoring interface. The system removes the client ID from a "pending download table" at the server. This only occurs when a "confirm download" message has been received by the server. The argument that removing a reference in a table stored at the server and deleting a unique ID at the monitoring interface would be the same is not a valid argument. *Birdwell* uses these IDs in the pending download table to ensure that the data has been downloaded. As Appellants previously stated regarding the instant application, it is not important to confirm that any data has been downloaded; rather, all that is required is that the system be set up in a reliable manner such that software vendors, once they provide a code to a user, are confident that the user, the ones paying for this code, can then download the information on the broadcast that is provided on the

[163] See *Birdwell* Col. 5, lines 14-15.
[164] See *Birdwell* Col. 6, lines 39-54.

broadcast channel.[165] The primary concern of any software vendor is that, once an authorization code is provided, only one copy of the software could be downloaded and not to allow the download of multiple copies. *Birdwell* is not concerned as to whether multiple copies are downloaded but, rather, wants to ensure that the server is efficient by not transmitting the data more times than is necessary. As such, *Birdwell* does not show any way to delete the unique ID from the monitoring interface such that information is prevented from being downloaded multiple times. In fact, *Birdwell* clearly allows multiple broadcasts to a particular system when it is running due to the fact that the confirm operation is not necessarily connected to the download operation. Again, this is set forth in Col. 7, lines 24-25. Further, in Col. 5, beginning at line 4, it is set forth as follows:

> In a preferred embodiment, a client computer system can receive download data for a client APP even though the client APP is not currently executing and connected to the server APP. The client FTC and client DDC control the receiving of download data. If download data is received when the client APP is not executing, the data is stored until the client APP starts execution and retrieves the data.[166]

Thus, confirmation of the download will not occur until a later time in the event that this is the case.

The Examiner identified a particular element in the prior art, that being the limitation of deleting a unique ID, as inferred from the removing of a client ID reference from a table stored at the server. *Kahn* stated that "a mere identification in the prior art of each element is insufficient to defeat the patentability of the combined subject matter as a whole."[167] Rather than concentrate on this element, the Examiner is required to articulate the basis on which the Examiner concludes that it would have been obvious to make the claimed invention, i.e., why one of ordinary skill in the art would have been motivated to select the references and to combine them in order to render the claimed invention obvious. The Examiner's indication that an ID exists and can be removed from a table at the server does not show the existence of such teaching. Thus, the Examiner has not met a *prima facie* case by stating, "It would have been obvious to one of

[165] See Response dated July 14, 2005, page 14.
[166] See Birdwell Col. 5, lines 4-11.
[167] *Kahn*, 441 F.3d at 986.

ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Tarr regarding software download to a set-top box with the teachings of Birdwell regarding deleting ID's *(sic)* after downloading is complete because deleting already used ideas would save space in memory."[168]

Birdwell contains no teaching, suggestion, or motivation to provide "one or more discrete data streams having a unique ID associated thereweith, which unique ID for each of the one or more discrete data streams is associated therewith by an associated software vendor and each of the unique IDs is unique to the user, and wherein said unique ID for each of the one or more discrete software data streams downloaded is first stored in a user storage device then deleted from the user storage device after downloading of the one or more discrete data streams. Any ID of *Birdwell* is only removed from a table stored *at the server* after download confirmation message is received by the server, therefore, reliance on this one particular aspect is insufficient to show any motivation, suggestion, or teaching that would lead one skilled in the art at the time of the invention to combine the teachings of *Birdwell* with *Tarr* to allow one with the teaching of *Tarr* in front of them to incorporate a unique ID stored by a user in a monitoring interface of the user, wherein the unique ID is deleted from the monitoring interface after download of software.

3. Conclusion – TSM Test.

Although the recent *KSR* Supreme Court case has indicated that the teaching-suggestion-motivation (TSM) test is not a rigid test, it is still considered to be a factor. Under this test, there must be some type of teaching in each of the references for combination as well as some kind of suggestion. There also must be some motivation to combine the two references. If this test alone were utilized, the question would be whether there is any teaching in *Tarr* and *Birdwell* that would suggest to one skilled in the art to combine the two references or is there any motivation to so combine.

Tarr is a reference that provides a start-up CD-ROM to initialize the system. The CD-ROM connects to a secondary processor, which is essentially the gaming system. The CD-ROM contains operating instructions, an encryption/decryption algorithm, and graphics that enable a primary processor to select and download gaming software for various game platforms. A client

[168] See Final Office action mailed September 06, 2006, page 3 paragraph 3.

must obtain a CD-ROM that is compatible with the game platform in his/her home. The user selects a game for download from a list provided by a head-end server. The user may select multiple games for download. The user also may select a previously downloaded game to be downloaded again. If the downloads are encrypted, the CD-ROM provides decryption. In this particular example, there is no unique ID associated with the software data streams and unique to the user. Additionally, in the *Tarr* system the user selects the game from a list such that the only software data steams sent to the user are those that correspond to those that the user selects. As such, all the software data streams broadcast to the user are stored. *Tarr* contains no suggestion or teaching that a filtering operation based on a unique ID occurs.

Birdwell provides a system that opportunistically downloads data to multiple computers. A central server assigns ID's to the computer systems of the clients. The ID's are not assigned by the software vendors. When a recipient requests the data, the server sends the data to all the clients on a distribution list. The server also stores the client IDs in a "pending download" table. Each client sends a confirm download message to the server when it receives the data. When the server receives the confirm download message, the server removes the clients ID from the "pending download" table. However, the server repeatedly sends the data to a client until that client sends back the confirm download message. There is no indication that this client ID is stored in the user system or even that the ID is deleted from the server. The server only removes the client ID from a table stored on the server after the server receives a confirm download message. The claims require that the unique ID be deleted from the user storage device after downloading of the software data stream. Thus, *Birdwell* contains no teaching therein for the type of unique ID, unique to a user, that is associated by an associated vendor and where the unique ID is deleted from the user storage device after downloading of the software data stream.

Therefore, no reason, motivation or suggestion exists to combine *Tarr* with *Birdwell*. *Tarr* has no need to use the IDs in the system of *Birdwell*, as the *Tarr* system provides a CD-ROM that is required for downloading of software. Since the IDs used in *Birdwell* are used at the server to confirm that the software has been downloaded, the question is "Why would one skilled in the art want to delete an ID on a CD-ROM for the purpose of allowing an authorized user to download a software data stream only once?" Doing so would be no different than a

software vendor providing the software data as part of the CD-ROM. As such, there is no motivation or suggestion that would in any way lead one skilled in the art to combine such.

Based on the TSM test, the Examiner's position is conclusory. The Examiner states that the combination of *Tarr* and *Birdwell* would provide a unique ID, associated to one or more discrete data streams, which unique ID of each of the one or more discrete data streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user. The Examiner further states that this combination would "provide a unique ID that is deleted from a user storage device after downloading of the one or more discrete data streams because deleting already used ideas would save space in memory." However, the Examiner has provided no articulated reasoning why deleting this unique ID from a CD-ROM would save memory. Neither *Tarr* nor *Birdwell* contain a unique ID that is unique to the user, which unique ID is associated by an associated software vendor and where the unique ID is stored on a user interface by a user. Therefore, neither reference deletes the unique ID from the user interface.

4. *KSR* Test:

The recent *KSR* case, although not fully analyzed as to its impact on obviousness type rejections under 35 U.S.C. § 103, indicates that the test is "if a person of ordinary skill can not implement a *predictable variation*, §103 likely bars it's patentability."[169] Under this dictum, the question would be whether *Tarr* could be varied in a predictable manner to remove an ID stored in pending download table to grant an authorized user access to download a software data stream only once. *Tarr* would have no benefit to have a unique ID that confirms that a download operation occurred. In Claim 1, the purpose of the ID is to associate the software data stream with a user, filter the software data stream, and provide the user limited access to the data stream by deleting the ID from the user system after download. If a unique ID to confirm that download occurred were used in the *Tarr* system, there is no indication a filtering operation would occur, nor is there any indication the authorized user would only be allowed a single download operation. As such, there is no predictable variation of *Tarr* that would lead one skilled in the art to utilize the *Birdwell* method of removing an ID upon receipt of a confirm download message. When work is available in one field of endeavor, i.e., downloading software data streams to authorized users, there is no design incentive or other market force that would prompt a

[169] *KSR,* 127 S. Ct. at page 1740.

predictable variation of the *Tarr* reference to utilize a unique ID for a purpose that is not useful or envisioned in *Tarr*. In summary, Appellants submit that the Examiner has failed to provide a *prima facie* case as to why the *Tarr* and *Birdwell* references, in combination, obviate Appellants' present inventive concept, as defined by claims 1-30.

D. Dependent Claims 3, 4, 18 and 19 as rejected by the combination of *Tarr*, *Birdwell* and *Metz*.

In the Final Office Action mailed August 24, 2006, the Examiner maintains his 35 U.S.C. § 103 rejection of Claims 3, 4, 18 and 19. On page 4, paragraph 4 of the Final Office Action the Examiner states "[as] to claim 18, the teaching of the *Tarr-Birdwell* combination combine to make the teaching of claim 16 obvious; however, they do not explicitly teach system where one or more software data streams are checked for errors when being downloaded." As stated hereinabove, the combination of *Tarr* and *Birdwell* does not disclose one or more discrete data streams having a unique ID associated therewith, which unique ID is associated therewith by an associated software vendor and each of the unique IDs is unique to a user, and wherein the user first stores each of the unique IDs to a user interface, the unique ID for each of the one or more discrete data streams is deleted from the user interface after download. Claims 3 and 4 depend from and further limit Claim 1 while Claims 18 and 19 depend from and further limit Claim 16. As such, Claims 3, 4, 18, and 19 are allowable for at least the same reasons as the claims from which they depend, as discussed above.

The Examiner has provided *Metz* to cure the deficiencies in *Tarr-Birdwell*. Specifically, the Examiner has relied on *Metz*, "[to teach] a system wherein the one or more software data streams are checked for errors being downloaded (col. 37, lines 44-59)."[170] The Examiner further states:

> It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of the Tarr-Birdwell regarding a system for downloading software with the teachings of Metz regarding checking for errors because a downloaded data stream would be useless if it had errors.[171]

[170] See Final Office action mailed September 06, 2006, page 4 paragraph 11.
[171] See Final Office action mailed September 06, 2006, page 4 paragraph 11.

The primary purpose of *Metz* is to allow a software vendor to be able to provide upgrades to software once the software is distributed to set-type terminal devices[172]. To facilitate this, a broadcast channel is defined that repeatedly broadcasts upgrades for the software.[173] The system formats the data into individual blocks for download.[174] The microprocessor of the set-type terminal device captures the entire data file in sequence. The microprocessor may also capture the entire data file out of sequence. In the later case, the microprocessor rearranges the blocks to compile the data in the proper sequence.[175] The microprocessor loads the captured data into volatile RAM.[176] After the microprocessor loads the captured data into volatile RAM, the microprocessor performs a checksum operation on the data. The checksum operation determines if any errors are present in the received data.[177] If an error is present in the data, the microprocessor downloads the data again.[178]

Metz does not cure the deficiencies of *Tarr-Birdwell*. Dependent Claim 3 requires an error checking step *during* the step of downloading such that one or more software data streams which are downloaded are checked for errors *during the download process*. The Examiner cited *Metz* at column 37, lines 44-59 to support the 35. U.S.C. § 103(a) rejection. The specific portion of the disclosure sets forth:

> The DET microprocessor 110 loads the captured operating system into volatile RAM 122. The DET microprocessor 105 then performs a checksum operation on the data file to determine if there are any errors in the received data (step S6). If the checksum result is not valid, indicating errors in the extracted copy of the operating system *stored* in the RAM 122, then the microprocessor 110 returns to step S5 and again extracts the relevant operating system file from the broadcast carousel. Although the separate steps are not illustrated, the microprocessor 110 will repeat steps S5 and S6 up to some predetermined number of times. If extraction is not successful as indicated by a valid checksum in step S6 by the predetermined number of attempts, the microprocessor will terminate running of the upgrade routine and

[172] See *Metz* Abstract.
[173] See *Metz* Abstract
[174] See *Metz* Col. 37, lines 7-8.
[175] See *Metz* Col. 37, lines 8-15.
[176] See *Metz* Col. 37, lines 15-45.
[177] See *Metz* Col. 37, lines 45-48.
[178] See *Metz* Col. 37, lines 48-59.

> will reboot the existing operating system still stored in the NVRAM 121.[179] (*emphasis added*).

Metz does teach a checksum process. However, the checksum process of *Metz* does not occur until *after* download of data is complete. As such, *Metz* does not teach or suggest an error checking step *during* the step of downloading such that one or more software data streams which are downloaded are checked for errors *during the download process*.

E. Dependent Claims 6, 12-14, 21 and 27-29 as rejected by the combination of *Tarr*, *Birdwell* and *Brandenburg*.

The Examiner has provided *Brandenburg* to cure the deficiencies of *Tarr-Birdwell*. Claims 6, 12, 13 and 14 depend from and further limit Claim 1. Claims 21 and 27, 28 and 29 depend from and further limit Claim 16. These dependent claims are allowable for at least the same reasons the claims from which they depend as discussed above. The addition of *Brandenburg* does not cure the deficiencies of *Tarr-Birdwell*.

F. Dependent Claims 7, 8, 22 and 23 as rejected by the combination of *Tarr*, *Birdwell* and *Durden*.

Claims 7 and 8 depend from and further limit Claim 1. Claims 22 and 23 depend from and further limit Claim 16. These dependent claims are allowable for at least the same reasons the claims from which they depend as discussed above. The addition of *Durden* does not cure the deficiencies of *Tarr-Birdwell*.

G. Dependent Claims 9 and 24 as rejected by the combination of *Tarr*, *Birdwell*, *Durden* and *Fries*.

Claims 7 depends from and further limits Claim 1. Claims 24 depends from and further limits Claim 16. These dependent claims are allowable for at least the same reasons the claims from which they depend as discussed above. The addition of *Fries* does not cure the deficiencies of *Tarr-Birdwell-Durden*.

VIII. Conclusion

In Summary, Appellants submit that these references fail to provide a suggestion, motivation, or teaching for the various combinations because the text fails to illustrate "why" one

[179] See *Metz* Col. 37, lines 44-59.

skilled in the art would combine the references in the particular manner required to provide a predictable variation. Instead, the Examiner simply identifies particular components for each reference, combines them in a specific manner required by Appellants' claimed invention, and then states that it would be obvious to one skilled in the art to do so. This is clearly hindsight based reasoning that contravenes the standards imposed by both the MPEP and the Federal Circuit, and Appellants respectfully submit that the cited combinations are improper for reasons detailed above and requests that the rejections under § 103 be withdrawn.

Respectfully submitted,

HOWISON & ARNOTT, L.L.P.

/Gregory M. Howison, Reg. # 30,646/

Gregory M. Howison
ATTORNEYS FOR THE APPELLANTS
P.O. Box 741715
Dallas, TX 75374-1715
September 24, 2007

CLAIMS APPENDIX

1. (Previously Presented): A method for distributing software, comprising the steps of:

providing a television broadcast distribution system having one or more broadcast channels for broadcasting analog and digital television information to a receiver of a user;

designating select ones of the one or more broadcast channels for the transmission of one or more discrete software data streams;

transmitting the one or more discrete software data streams over the select ones of the one or more broadcast channels at a scheduled time, each of the one or more discrete software data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user;

the user associating with a monitoring interface of the user that is connected to the receiver, the unique ID associated with that user and a desired one of the one or more discrete software streams;

selecting, in the monitoring interface associated with the user, selected ones of the one or more discrete software data streams according to the respective unique IDs for download via the receiver;

downloading the selected one or more discrete software data streams to a user storage device during the scheduled time for use by the user, the user storage device connected to the receiver through said monitoring interface; and

deleting the unique ID from the monitoring interface for each of the one or more discrete software data streams downloaded after downloading thereof.

2. (Original): The method of Claim 1, wherein the television broadcast distribution system is a cable television broadcast system.

3. (Original): The method of Claim 1, further comprising an error checking step during the step of downloading such that the one or more software data streams which are downloaded are checked for errors during the download process.

4. (Original): The method of Claim 3, wherein the receiver automatically re-selects for download, and downloads, the one or more software data streams which fail the step of error checking.

5. (Original): The method of Claim 1, wherein the step of broadcasting broadcasts the one or more software data streams repetitively during a specific period of time.

6. (Original): The method of Claim 1, wherein the step of broadcasting broadcasts the one or more software data streams once during a specific time period.

7. (Original): The method of Claim 1, further comprising a step of accounting which logs the unique IDs of the one or more software data streams which were downloaded with an accounting device, and transmits the unique IDs to a provider of the one or more software data streams using the accounting device.

8. (Original): The method of Claim 7, wherein the accounting device interfaces to a public-switched telephone network, and transmits the unique IDs over the public-switched telephone network to the provider of the one or more software data streams.

9. (Original): The method of Claim 7, wherein the accounting device interfaces to a packet-switched global communication network, and transmits the unique IDs over the global communication network to the provider of the one or more software data streams.

10. (Original): The method of Claim 1, wherein the step of selecting further comprises the step of programming the receiver with time, channel, and the unique ID information.

11. (Original): The method of Claim 1, wherein the one or more software data streams comprise software applications which are broadcast on a first channel, and one or more software updates which are broadcast on a second channel.

12. (Original): The method of Claim 1, wherein the step of downloading downloads the selects ones of the one or more software data streams directly to a user computer over a communication link existing between the receiver and the user computer.

13. (Original): The method of Claim 12, wherein the communication link is a universal serial bus.

14. (Original): The method of Claim 12, wherein the communication link is a high-performance serial bus.

15. (Original): The method of Claim 1, wherein the television broadcast distribution system is a digital television broadcast system.

16. (Previously Presented): A system for distributing software, comprising:

a television broadcast distribution system having one or more broadcast channels for broadcasting analog and digital television information to a receiver of a user;

one or more discrete software data streams designated for transmission on select ones of said one or more broadcast channels, each of the one or more discrete data streams having a unique ID associated therewith, which unique ID for each of the one or more discrete software streams is associated therewith by an associated software vendor and each of the unique IDs is unique to a user; and

a user storage device associated with the user and connected to said receiver through a monitoring interface and said user storage device operable to store both one or more of the unique IDs associated with the user prior to downloading of the one or more discrete software data streams and, after downloading thereof, for storing said downloaded one or more discrete software data streams-wherein each of said unique IDs stored in said user interface is stored there by the user;

wherein said one or more discrete software data streams are transmitted over said select ones of said one or more broadcast channels at a scheduled time, each of said one or more discrete software data streams having associated therewith the associated one of said unique IDs; and

wherein said select ones of said one or more discrete software data streams are downloaded via said receiver to said monitoring interface for filtering said discrete software data streams according to said respective unique IDs, and wherein said unique ID for each of the one or more discrete software data streams downloaded deleted from the user storage device after downloading thereof.

17. (Original): The system of Claim 16, wherein said television broadcast distribution system is a cable television broadcast system.

18. (Original): The system of Claim 16, wherein said select ones of said one or more software data streams are checked for errors when being downloaded.

19. (Original): The system of Claim 18, wherein said receiver automatically re-selects for download, and downloads, said one or more software data streams which fail the error checking process.

20. (Original): The system of Claim 16, wherein said one or more software data streams are broadcast repetitively during a specific period of time.

21. (Original): The system of Claim 16, where said one or more software data streams are broadcast once during a specific time period.

22. (Original): The system of Claim 16, wherein an accounting device logs said unique IDs of said one or more software data streams which were downloaded, and transmits said unique IDs to a provider of said one or more software data streams using said accounting device.

23. (Original): The system of Claim 22, wherein said accounting device interfaces to a public-switched telephone network, and transmits said unique IDs over said public-switched telephone network to said provider of said one or more software data streams.

24. (Original): The system of Claim 22, wherein said accounting device interfaces to a packet-switched global communication network, and transmits said unique IDs over said global communication network to said provider of said one or more software data streams.

25. (Original): The system of Claim 16, wherein said receiver is programmed by inputting parameters which comprise time, channel, and said unique ID information.

26. (Original): The system of Claim 16, wherein said one or more software data streams comprise software applications which are broadcast on a first channel, and one or more software updates which are broadcast on a second channel.

27. (Original): The system of Claim 16, wherein said select ones of said one or more software data streams are downloaded directly to a user computer over a communication link existing between said receiver and said user computer.

28. (Original): The system of Claim 27, wherein said communication link is a universal serial bus.

29. (Original): The system of Claim 27, wherein said communication link is a high-performance serial bus.

30. (Original): The system of Claim 16, wherein said television broadcast distribution system is a digital television broadcast system.

EVIDENCE APPENDIX

A. U.S. Patent No. 6,238,290 to Tarr et al. ("*Tarr*") found in paragraphs 2-21 and 23-25 of the Final Office Action (mailed September 8, 2006); paragraphs 2-21 and 23-25 of the Office Action (mailed October 20, 2005); paragraphs 2-4, 11-19 and 22-24 of the Final Office Action (mailed February 25, 2003); and page 5 of the Office Action (mailed July 26, 2002).

B. U.S. Patent No. 6,002,852 to Birdwell et al. ("*Birdwell*") paragraphs 2-3, 10-11, 14-21 and 24-25 of the Final Office Action (mailed September 8, 2006); paragraphs 2-3, 10-11, 14-21 and 24-25 of the Office Action (mailed October 20, 2005); paragraphs 2-3 and 12-25 of the Office Action (mailed February 17, 2005); paragraphs 3, 5, 7, 9 and 11 of the Final Office Action (mailed May 3, 2004); and paragraphs 3, 4 and 13-24 of the Office Action (mailed December 4, 2003).

C. U.S. Patent No. 5,666,293 to Metz et al. ("*Metz*") paragraphs 10-12 of the Final Office Action (mailed September 8, 2006); paragraph 10 of the Office Action (mailed October 20, 2005); paragraphs 2-10, 12-19 and 22-23 of the Office Action (mailed February 17, 2005); paragraphs 3, 5, 7, and 9 of the Final Office Action (mailed May 3, 2004); paragraphs 3-6 and 13-24 of the Office Action (mailed December 4, 2003); paragraphs 2-3 and 5-19 of the Final Office Action (mailed February 25, 2003); and pages 2-7 of the Office Action (mailed July 26, 2002).

D. U.S. Patent No. 5,894,516 to Brandenburg ("*Brandenburg*") found in paragraphs 14-18 of the Final Office Action (mailed September 8, 2006); paragraphs 14-18 of the Office Action (mailed October 20, 2005); paragraphs 15-16 of the Office Action (mailed February 17, 2005); paragraph 5 of the Final Office Action (mailed May 3, 2004); paragraphs 13-18 of the Office Action (mailed December 4, 2003); paragraphs 12-24 of the Final Office Action (mailed February 25, 2003); and pages 3-7 of the Office Action (mailed July 26, 2002).

E. U.S. Patent No. 5,003,384 to Durden et al. ("*Durden*") found in paragraphs 20-25 of the Final Office Action (mailed September 8, 2006); paragraphs 20-25 of the Office Action (mailed October 20, 2005); paragraphs 18-23 of the Office Action (mailed February 17, 2005); paragraphs 7-9 of the Final Office Action (mailed May 3, 2004); paragraphs 19-24 of the Office

Action (mailed December 4, 2003); and paragraphs 18-24 of the Final Office Action (mailed February 25, 2003).

F. U.S. Patent No. 6,317,885 to Fries ("*Fries*") found in paragraphs 24-26 of the Final Office Action (mailed September 8, 2006); paragraphs 24-26 of the Office Action (mailed October 20, 2005); paragraph 9 of the Final Office Action (mailed May 3, 2004); paragraphs 22-24 of the Office Action (mailed December 4, 2003); and pages 6 and 7 of the Office Action (mailed July 26, 2002).

G. *KSR International Co. v. Teleflex Inc., et al.*, 127 S. Ct. 1727 (2007).

US006238290B1

(12) **United States Patent**
Tarr et al.

(10) **Patent No.: US 6,238,290 B1**
(45) **Date of Patent: *May 29, 2001**

(54) **SYSTEM AND METHOD FOR SCHEDULED DELIVERY OF A SOFTWARE PROGRAM OVER A CABLE NETWORK**

(75) Inventors: **Jeffrey R. Tarr**, Denver; **J. Clarke Stevens**, Broomfield; **Jeffrey C. Lutz**, Niwot, all of CO (US)

(73) Assignees: **Mediaone Group, Inc.**, Englewood; **U S West, Inc.**, Denver, both of CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/315,767**

(22) Filed: **May 20, 1999**

Related U.S. Application Data

(63) Continuation of application No. 08/520,663, filed on Aug. 28, 1995, now Pat. No. 5,935,004.

(51) **Int. Cl.**[7] **A63F 9/22**
(52) **U.S. Cl.** **463/40**
(58) **Field of Search** 463/29, 40, 41, 463/42; 348/6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20; 455/402

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,319,705 6/1994 Halter .

FOREIGN PATENT DOCUMENTS

5-115063 9/1975 (JP) .
7-73124 3/1995 (JP) .
7-93148 4/1995 (JP) .

OTHER PUBLICATIONS

Minori Torii, Notice of Reason for Rejection, Japanese Office Action, Aug. 4, 1998.

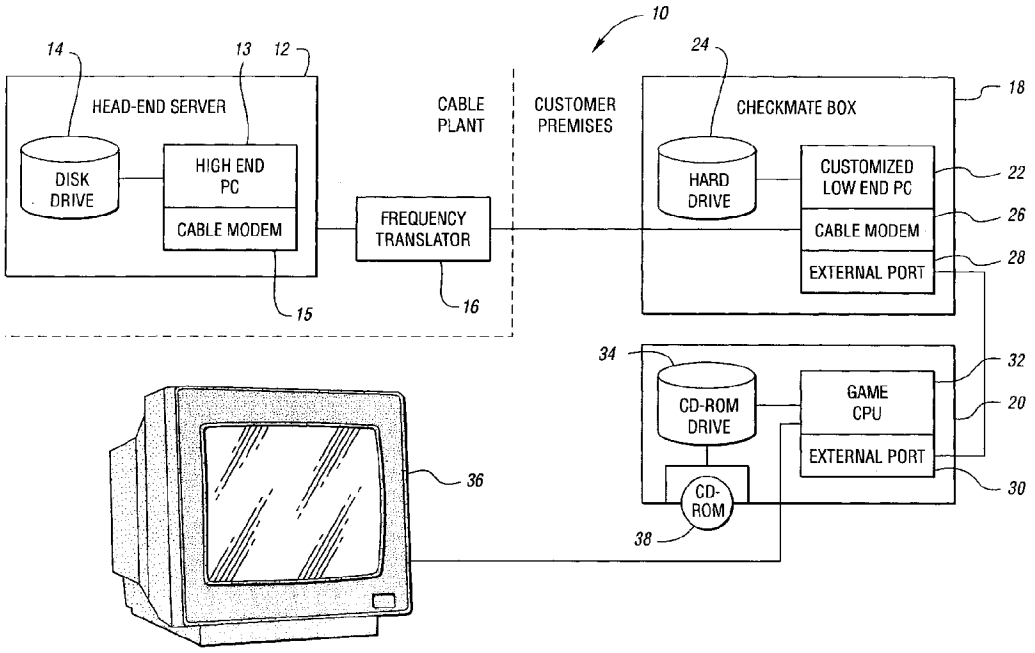
Minori Torii, Notice of Reason for Rejection, Japanese Office Action, Feb. 4, 1998.

Primary Examiner—Michael O'Neill
(74) *Attorney, Agent, or Firm*—Brooks & Kushman P.C.

(57) **ABSTRACT**

A system and method for delivering a software program over a cable network includes a head-end server operative to compress a plurality of software programs and transmit the compressed software programs over the cable network at a plurality of corresponding predetermined scheduled times. A primary processor in communication with the cable network receives a selected one of the plurality of software programs at the corresponding predetermined scheduled time, decompresses the program and stores it. A secondary processor in communication with the primary processor includes navigational software and drivers to enable the primary processor to select and receive the software program from the cable network without upstream communication to the head-end server.

1 Claim, 4 Drawing Sheets



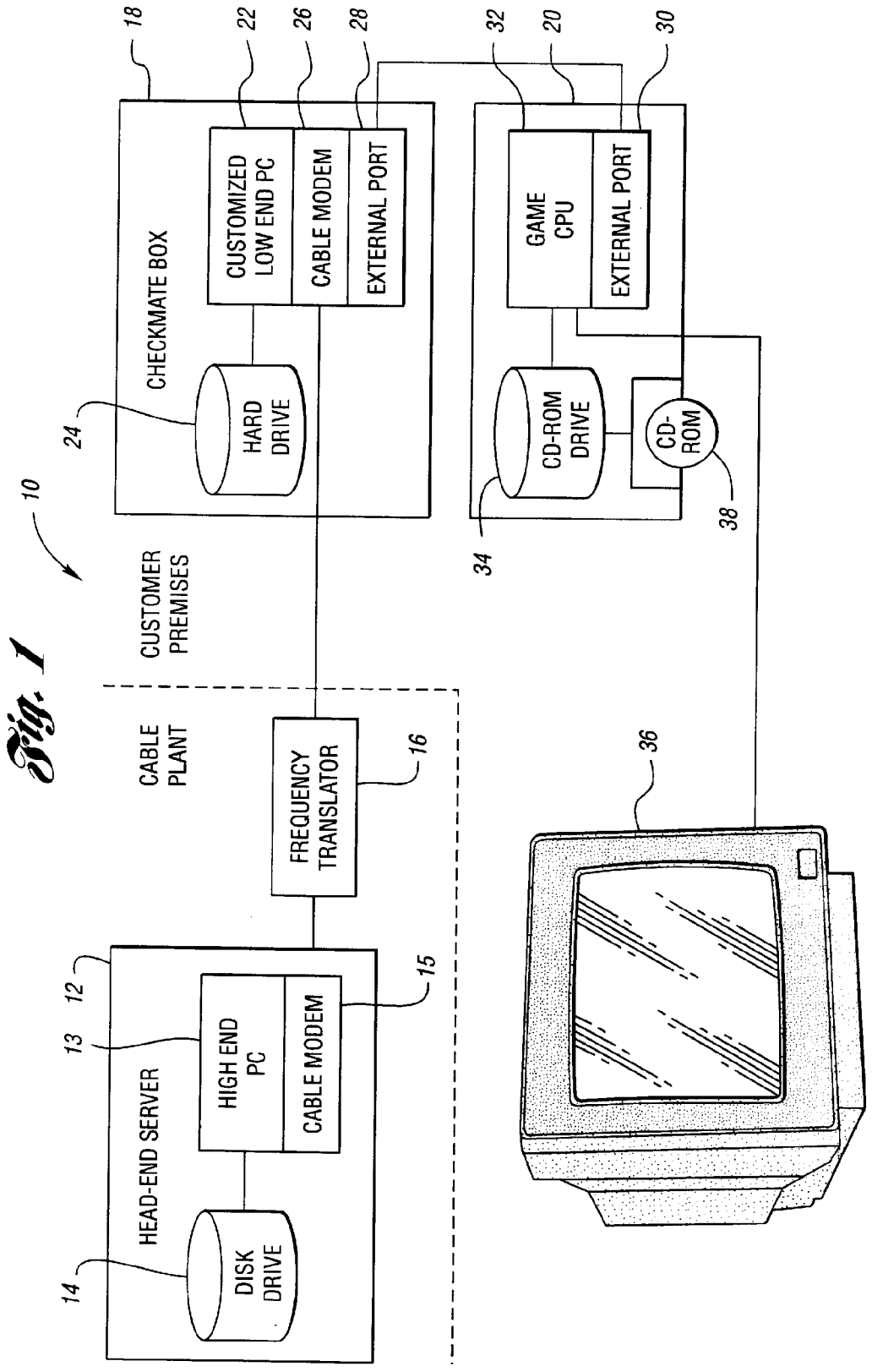

10
CUSTOMER PREMISES
CABLE PLANT
CHECKMATE BOX
CUSTOMIZED LOW END PC
CABLE MODEM
EXTERNAL PORT
HARD DRIVE
GAME CPU
EXTERNAL PORT
CD-ROM DRIVE
CD-ROM
FREQUENCY TRANSLATOR
HEAD-END SERVER
HIGH END PC
CABLE MODEM
DISK DRIVE
18
22
26
28
32
20
30
24
34
38
16
36
12
13
15
14

Fig. 1

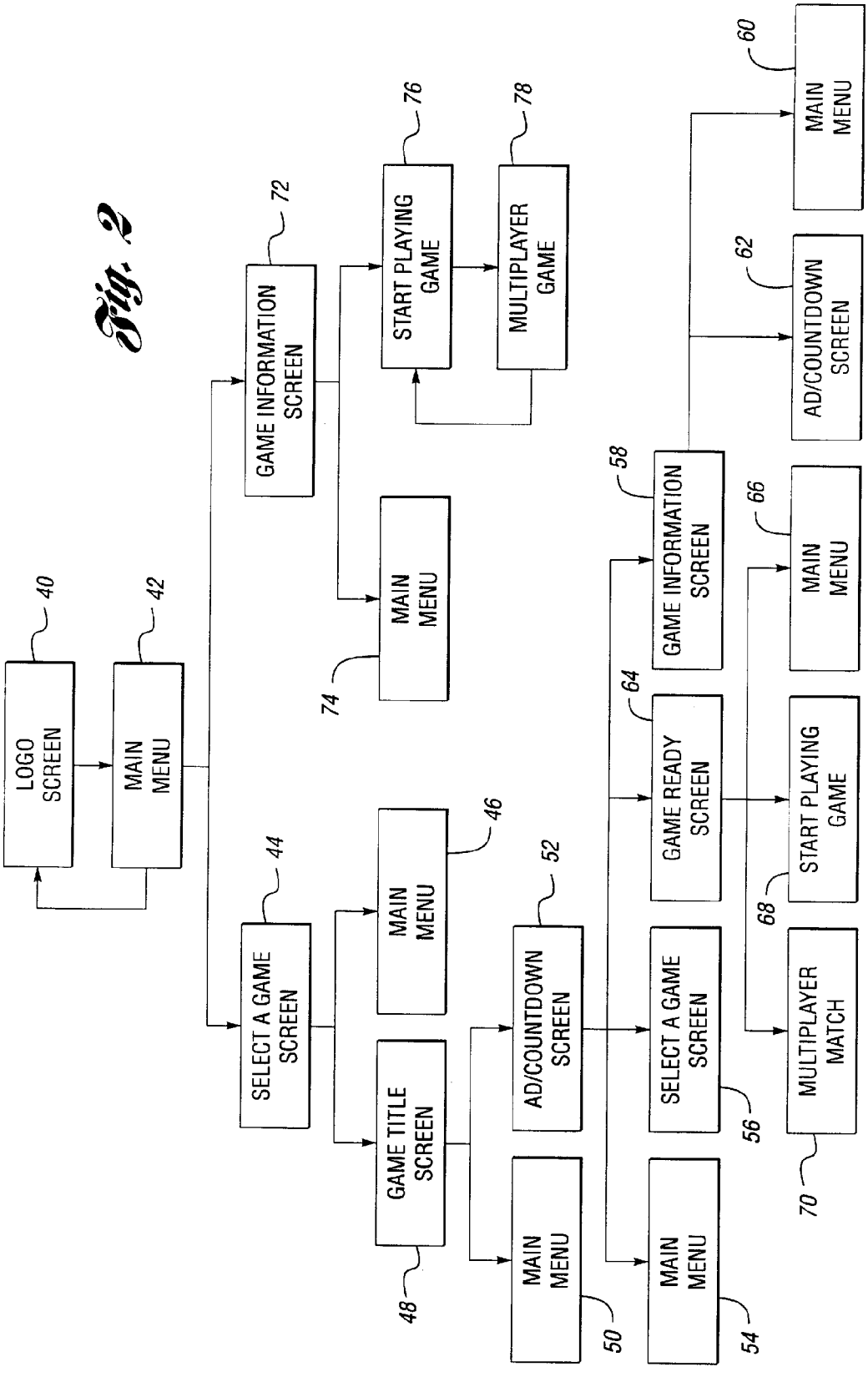

LOGO SCREEN
40
MAIN MENU
42
SELECT A GAME SCREEN
44
GAME TITLE SCREEN
48
MAIN MENU
46
MAIN MENU
50
AD/COUNTDOWN SCREEN
52
SELECT A GAME SCREEN
56
MAIN MENU
54
GAME READY SCREEN
64
MULTIPLAYER MATCH
70
START PLAYING GAME
68
MAIN MENU
66
GAME INFORMATION SCREEN
58
AD/COUNTDOWN SCREEN
62
MAIN MENU
60
GAME INFORMATION SCREEN
72
MAIN MENU
74
START PLAYING GAME
76
MULTIPLAYER GAME
78

Fig. 2

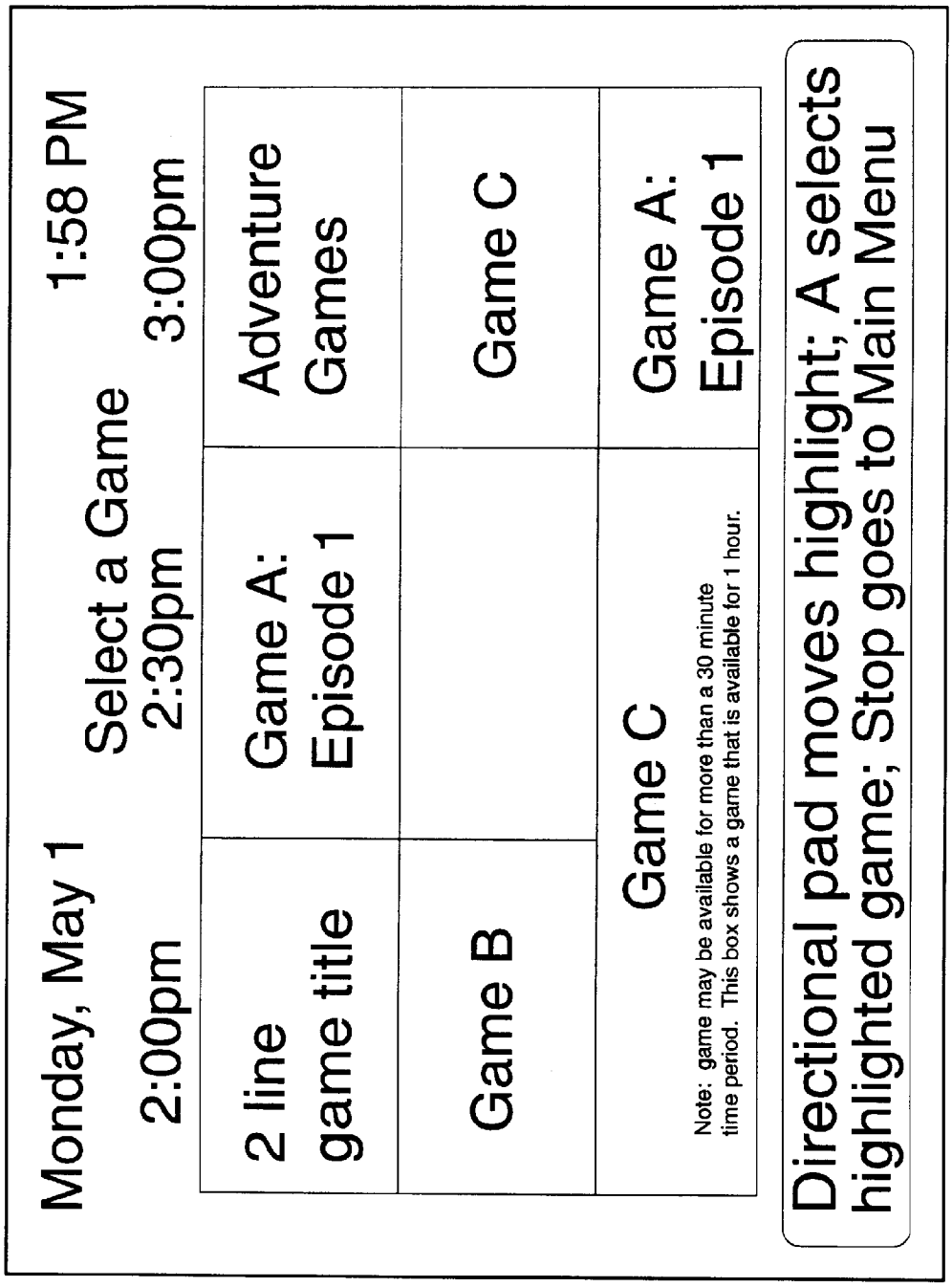
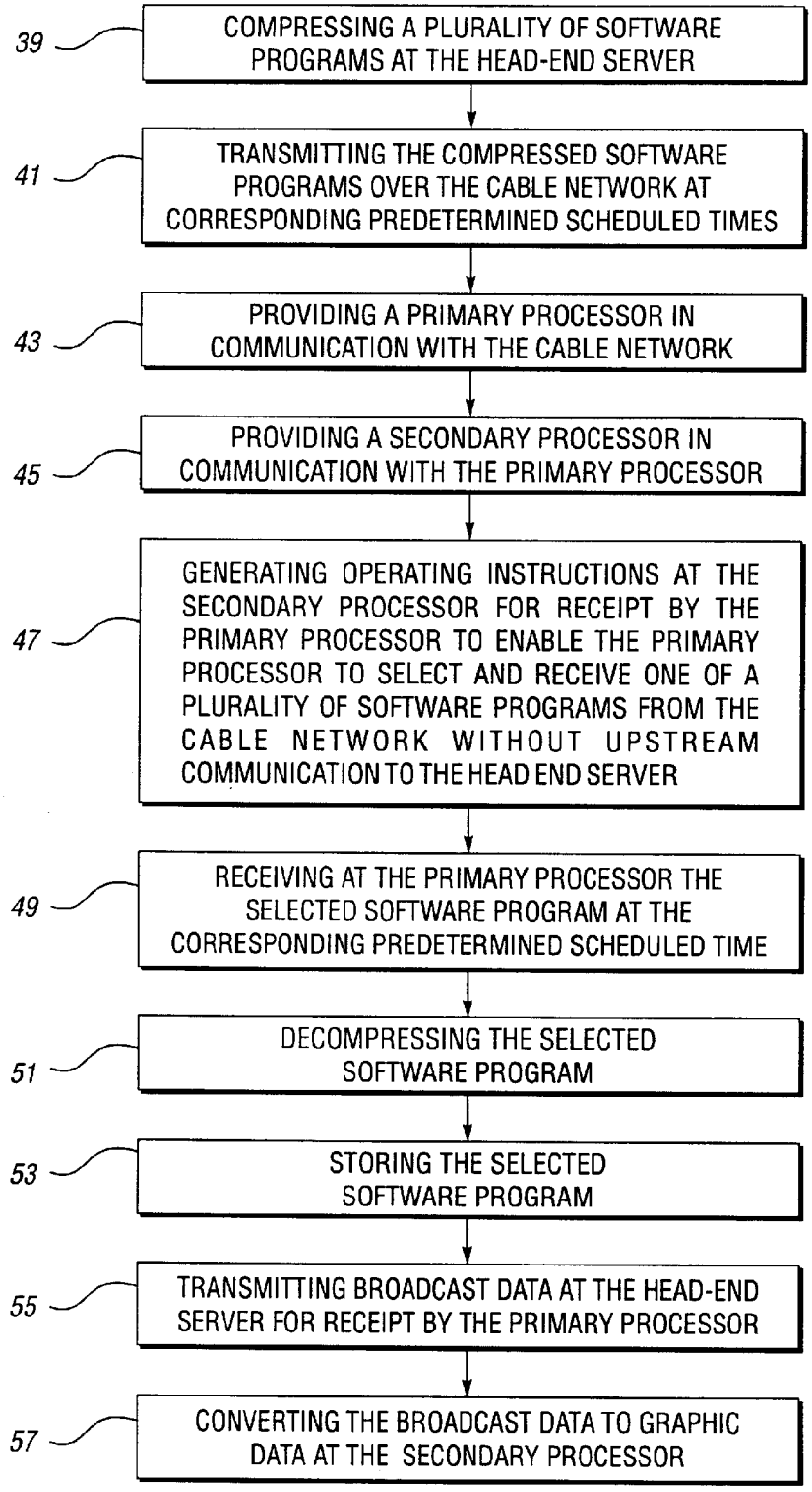

Monday, May 1
1:58 PM
Select a Game
2:00pm
2:30pm
3:00pm
2 line game title
Game A: Episode 1
Adventure Games
Game B
Game C
Game C
Note: game may be available for more than a 30 minute time period. This box shows a game that is available for 1 hour.
Game A: Episode 1
Directional pad moves highlight; A selects highlighted game; Stop goes to Main Menu

Fig. 3

Fig. 4


39
COMPRESSING A PLURALITY OF SOFTWARE PROGRAMS AT THE HEAD-END SERVER
41
TRANSMITTING THE COMPRESSED SOFTWARE PROGRAMS OVER THE CABLE NETWORK AT CORRESPONDING PREDETERMINED SCHEDULED TIMES
43
PROVIDING A PRIMARY PROCESSOR IN COMMUNICATION WITH THE CABLE NETWORK
45
PROVIDING A SECONDARY PROCESSOR IN COMMUNICATION WITH THE PRIMARY PROCESSOR
47
GENERATING OPERATING INSTRUCTIONS AT THE SECONDARY PROCESSOR FOR RECEIPT BY THE PRIMARY PROCESSOR TO ENABLE THE PRIMARY PROCESSOR TO SELECT AND RECEIVE ONE OF A PLURALITY OF SOFTWARE PROGRAMS FROM THE CABLE NETWORK WITHOUT UPSTREAM COMMUNICATION TO THE HEAD END SERVER
49
RECEIVING AT THE PRIMARY PROCESSOR THE SELECTED SOFTWARE PROGRAM AT THE CORRESPONDING PREDETERMINED SCHEDULED TIME
51
DECOMPRESSING THE SELECTED SOFTWARE PROGRAM
53
STORING THE SELECTED SOFTWARE PROGRAM
55
TRANSMITTING BROADCAST DATA AT THE HEAD-END SERVER FOR RECEIPT BY THE PRIMARY PROCESSOR
57
CONVERTING THE BROADCAST DATA TO GRAPHIC DATA AT THE SECONDARY PROCESSOR

SYSTEM AND METHOD FOR SCHEDULED DELIVERY OF A SOFTWARE PROGRAM OVER A CABLE NETWORK

This Application is a continuation of Ser. No. 08/520,663 filed Aug. 28, 1995 now U.S. Pat. No. 5,935,004.

TECHNICAL FIELD

This invention relates to interactive data services for communication networks and, in particular, to the scheduled broadcast of software programs to subscribers over a cable network.

BACKGROUND ART

Prior art video game players are cartridge-based systems wherein individual games are stored on ROM chips within corresponding cartridges. Typical cartridge-based game systems include, for example, Sega Genesis, Super Nintendo, etc. Each game can be played over and again simply by inserting the cartridge in the game player and following the required steps for start-up. These systems suffer, however, in that the cartridges are relatively expensive to make and have limited storage capability. Typical video games cartridges store files on the order of 1–3 megabytes.

In an effort to provide greater game selection at reduced cost, network providers have turned their attention toward the development of interactive data systems whereby video games may be transmitted to subscribers over existing cable networks. Prior art implementations of these systems broadcast game software electronically over coaxial cable plant as well as other broadband wireline and wireless networks. The games are received and stored on DRAM in a device which plugs into a subscriber's game player. Like the prior art cartridge-based systems, however, these network implementations are limited by the storage capability of the DRAM-based storage device which is on the order of 1–3 megabytes. Moreover, because the DRAM is volatile, video games can only be played while the system is powered. Once power is removed, the video game is lost.

A typical prior art network implementation of these DRAM-based systems is the Sega channel which delivers games of 3 megabytes or less. In operation, up to 50 games (totaling approximately 100 megabytes) are broadcast in a continuous stream over the cable network, in a broadcast pattern which repeats approximately every minute. The typical spectrum utilized is 6 MHz. The subscriber is provided with a menu of 50 games which change on a monthly basis and from which the subscriber selects the game he or she wishes to play. The selected game is then downloaded within approximately one minute of selection and stored in DRAM, and played by the subscriber.

While the prior art network implementation works well for the current generation of cartridge-based games, it is entirely ineffective in delivering next-generation CD-ROM-based games which are typically hundreds of megabytes in size (up to 650 megabytes). For example, if 50 next-generation games were sought to be delivered on a continuous basis, thousands of MHz of spectrum would be required. (In contrast, the most advanced cable networks have a total of only 750 MHz of spectrum). With existing hardware, download time would increase from approximately one minute to 5 hours or more and the device which connects to the game player and stores the games in the home would be required to incorporate over 600 megabytes of DRAM at a cost of tens of thousands of dollars. Download time could, of course, be decreased through the use of a high speed modem. To have any appreciable effect, however, such a modem would be required to transmit and receive data at a rate of several thousands of Mbps. The most advanced cable modems presently under commercial development, however, do not exceed 27 Mbps.

Even if these obstacles could be overcome, however, such an implementation would still be subject to the failings of the prior art network systems in that the game would be lost if power were turned off. Moreover, each video game manufacturer would be required to have its own unique device for downloading and storing the game.

In a further attempt to avoid the above-noted problems, designers have sought to utilize massive servers and processors at the cable head-end and use an upgraded broadband digital two-way cable plant to deliver games interactively. Such systems are commonly referred to in the art as virtual CD-ROM approaches. In operation, a small piece of a game is essentially moved from the server to a subscriber set-top box and as the subscriber moves through the game, signals are sent from the set-top box to the server and the next piece of the game is downloaded.

As is readily seen, this approach suffers from many problems as well. Namely, most households (on the order of 70,000,000) have access to a one-way cable plant only. Few consumers have access to two-way cable plants (far less than 10,000). Thus, the approach would require substantial upgrades to the cable plant, head-end equipment and consumer premises equipment costing thousands of dollars per subscriber (approximately $100,000,000.00 per head-end). Even if this feat could be accomplished, however, the existing two-way cable plant is far too slow to deliver the most popular games. As those skilled in the art will recognize, fast twitch action and sports games—which account for approximately 80% of the market—require round-trip latency of 100 milliseconds or less. The existing two-way cable systems, in contrast, have a round-trip latency which is many times that. Still further, each consumer must have access to dedicated bandwidth, greatly limiting the number of simultaneous users. Games would also be required to be significantly rewritten so as to be converted from CD-ROM delivery to electronic delivery. Such rewriting would cause software developers to incur many tens of thousands of dollars in cost and many months of coding and testing for each game title.

Consequently, a need has developed for an improved system and method for delivering video games and other software to network subscribers over a cable network. Such a system should permit fast transmission and easy low cost storage of large data files (on the order 600–650 megabytes) using downstream signaling only. Such a system should not require substantially greater bandwidth than that used by current DRAM-based systems. It should provide for multiple use of a downloaded software and should accommodate multiple application types such as games, shopping, news, etc. Multiple hardware platforms should also be supported as the secondary processor. These platforms should include SEGA, Saturn™, Sony Playstation™, 3DO, MacIntosh, PCs, etc. A catalog of multiple applications should be presented to the customer for potential downloading at scheduled times through a graphical user interface. The cost of the head-end and customer premise equipment should be commercially reasonable. Still further, no cable system network upgrades should be required. Such a system and method should also work with modem speeds that are reasonably attainable with current technology. Moreover, the software delivered should not require a high investment in recoding.

DISCLOSURE OF THE INVENTION

It is thus a general object of the present invention to provide an improved system and method for a subscriber to select and receive a software program from a plurality of software programs transmitted over a cable network at corresponding predetermined scheduled times.

It is a further object of the present invention to provide such an improved system and method for selecting and receiving software programs having file sizes on the order of 600–650 megabytes without sacrificing bandwidth or download time and without requiring expensive network upgrades.

It is a further object of the present invention to provide such an improved system and method which allows for the selection, transmission and storage of software programs using downstream transmission only.

Still further, it is an object of the present invention to provide an improved system and method for selecting and receiving software programs of the type described above which does not require substantially greater bandwidth than that used by current DRAM-based systems and which further permits multiple use by subscribers for multiple applications on multiple game hardware systems and personal computers.

Yet still further, it is an object of the present invention to provide such an improved system and method which is compatible with modem speeds that are reasonably attainable with current technology and which does not require a significant investment in recoding.

In carrying out the above objects and other objects, features and advantages of the present invention, a method is provided for delivering a software program over a cable network to a subscriber. The cable network includes a head-end server located at the cable plant for compressing a plurality of software programs and transmitting the compressed software programs over the cable network at a plurality of corresponding predetermined scheduled times. A primary processor such as a low end PC (386 or 486 chip and motherboard) and hard disk are provided in communication with the cable network. A secondary processor such as a game machine having a game CPU and CD-ROM drive is provided in communication with the primary processor. Operating, i.e., navigation instructions are generated at the secondary processor for receipt by the primary processor so as to enable the primary processor to select and receive one of the plurality of software programs from the cable network without upstream communication to the head-end server. The selected software program is received at the primary processor at the corresponding predetermined scheduled time whereupon it is decompressed and stored.

In a preferred embodiment, broadcast data is transmitted at the head-end server for receipt by the primary processor. The broadcast data corresponds to the predetermined scheduled times the corresponding software programs are available to be received. The broadcast data is converted to executable code, graphic data, sound, etc. at the secondary processor for display on a television or display terminal.

A system is also provided for carrying out the steps of the above-described method. The system includes a head-end server which is located at the cable plant. The head-end server is operative to compress and transmit software programs over the cable network at corresponding predetermined scheduled times. A primary processor is provided in communication with the cable network for receiving a selected one of the plurality of software programs at its corresponding predetermined scheduled time and decompressing and temporarily storing the selected software program. A secondary processor in communication with the primary processor includes navigational software and drivers to enable the primary processor to select and receive the software program from the cable network without upstream communication to the head-end server. The system further includes a television set or display terminal in communication with the secondary processor to display the selected software program.

In a preferred embodiment, the primary processor includes a custom personal computer having a cable modem and a hard disk. Similarly, the secondary processor comprises a video game machine or personal computer which includes a Central Processing Unit (CPU) and a CD-ROM drive. The above-mentioned navigational software and drivers are delivered from a CD-ROM to the secondary processor and from the secondary processor to the primary processor.

The head-end server may also be adapted to transmit broadcast data for receipt by the primary processor. The broadcast data corresponds to the times that the corresponding software programs are available to be received. The secondary processor may also be adapted to include executable code and art for converting the broadcast data to graphic data for display on the television set or display monitor. Still further, the head-end server may be adapted to include encryption means to encode the software program before transmission. Likewise, the secondary processor may be adapted to include decryption means to decode the transmitted software programs upon receipt.

These and other objects, features, and advantages of the present invention are readily apparent from the following detailed description of the best modes for carrying out the invention when taken in connection with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of the system of the present invention;

FIG. **2** is a flow diagram of the user interface of the system of FIG. **1**;

FIG. **3** is a diagram of the "Select a Game" Screen Display of the user interface of FIG. **2**; and

FIG. **4** is a flow diagram of the method steps of the present invention.

BEST MODES FOR CARRYING OUT THE INVENTION

Turning now to FIG. **1**, there is shown a block diagram of the system of the present invention, designated generally by reference numeral **10**. The system **10** includes a head-end server **12** located at the head-end of a cable plant, Direct Broadcast Satellite (DBS) or broadband system. The head-end server **12** transmits software programs such as digitized video, video games, digital audio, or other software programs over the network. The software programs are transmitted in the form of digitally encoded data. Different programs are broadcast at predetermined scheduled times over a bandwidth of approximately 6 MHz (the offering could be expanded by using more spectrum). The head-end server **12** preferably includes a "high end", i.e., high speed processor **13**, such as an Intel® Pentium® processor, a disk drive **14** and a cable modem **15**.

The system **10** also includes a conventional frequency translator **16** coupled to the head-end server **12**. The fre-

quency translator **16** transfers a block of signals occupying a selected frequency band, such as the 6 MHz spectrum occupied by a typical cable television channel, from one position in the frequency spectrum to another. The digital data may be encoded onto an analog cable, e.g., a 75 ohm coaxial cable, in one of several ways, including Quadrature Phase Shift Keying (QPSK), Quadrature Amplitude Modulation (QAM), or Vestigial Sideband (VSB) modulation, etc.

The system **10** further includes a primary processor **18** coupled to the head-end server **12** via the frequency translator **16**. The primary processor **18** is located within a subscriber's home or environment. The primary processor **18** receives and demodulates the broadcast and stores the data on a hard disk, thus emulating the CD-ROM drive typically supported by a secondary processor **20**. The primary processor **18** preferably includes a "customized low end", i.e., low speed processor **22** such as an Intel® 386 or 486, a hard disk **24**, a cable modem **26** and an external port **28**. The cable modem **26** communicates with the cable modem **15** of the head-end server **12** via either one-directional or bi-directional communication. The hard disk **24** stores the data received by the cable modem **26** of the primary processor **18** for future use until new data is transmitted and received. The primary processor **18** may also include a built-in smart card reader (not shown) for purchase transactions, a narrow band modem (not shown) to access on-line services and to communicate transactions, and a keyboard (not shown).

The secondary processor **20**, also located within a subscriber's home or environment, is in electrical communication with the primary processor **18** via an external port **30** of the secondary processor **20**. The electrical communication may be either bus-based or parallel or any other suitable communication that can be supported by secondary processor **20** and primary processor **18**. The secondary processor **20** may be a personal computer or any commercially available game machine and includes a game Central Processing Unit (CPU) such as a processor **32** and a disk drive **34**, such as a CD-ROM drive. The system **10** further includes a television or display monitor **36** for displaying the video games or software programs to the subscriber.

Display **36** is in electrical communication with the secondary processor **20**.

To start up the system, the user inserts a start-up CD-ROM **38** into the CD-ROM drive **34**. The CD-ROM **38** contains the device drivers and software which enable the PC or video game player to interact with the primary processor **18**. The CD-ROM **38** also contains the software which enables the primary processor **18** to select and receive software programs from the head-end server **12**. To prevent unauthorized acquisition of the video games and other content, the CD-ROM **38** also contains a program to decrypt the broadcast programs. The CD-ROM **38** may also contain multimedia content (e.g., graphics, animations, video clips and audio clips), software tools, programs (e.g., navigator), and software engines which can be used alone or with data that is broadcast over the network.

The system **10** of the present invention has the advantage of being compatible with a plurality of video game systems that may be purchased by the subscriber such as Sega, Saturn™, Sony, Playstation™, Nintendo Ultra, Atari Jaguar, etc. Thus, the network provider is not required to develop or select and supply a particular video game hardware system. Start-up CD-ROMs **38** are available for each of the various platforms of video game players, e.g., Sega TM, Sony TM, 3DO TM, IBM PC TM, Macintosh TM, etc. The subscriber would need to obtain the CD-ROM **38** corresponding to the platform available at his/her home. The CD-ROM **38** contains the executable code and the art for an electronic program guide for the platform chosen by the subscriber. The subscriber would then only be able to select programs compatible with his/her video game player.

The head-end server **12** compresses the video games prior to being broadcast over the communications network. Decompression will then occur "on the fly" or on a data block basis on the hard disk **24** of the primary processor **18**. The compression/decompression provides greater capacity for the transmission of video games over the cable. Decompression "on the fly" or on a data block basis eliminates the need to accommodate space for simultaneously storing both the compressed and uncompressed data on the hard disk **24**.

Small amounts of critical data, including text and graphics, sound, etc., are interleaved with the delivery of video games and other interactive programming so that time critical data does not have to wait until the end of a game transmission. This is important because video game transmissions can take several minutes.

As described above, the software programs delivered by the head-end server **12** are transmitted at predetermined scheduled times. The programming information that needs to be inserted into the electronic program guide is broadcast periodically to inform the subscriber when the program is available for downloading. As time passes and new programs are scheduled, the information about these programs is broadcast and the information in the electronic program guide is updated.

System **10** may also be applied to advertising. The CD-ROM **38** may store a selection of backgrounds, characters, software engines and sounds. The network provider, working with an advertiser, can then broadcast instructions to the primary processor **18** over the communications network to select specific backgrounds, characters, software engines and sounds and create an interactive advertisement while minimizing the time and the spectrum which would otherwise be required to broadcast the advertisement in its entirety. The network provider and the advertiser can further customize the advertisement by broadcasting logos, icons or new art as required.

Turning now to FIG. **4**, the method of the present invention may be more particularly described. As indicated above, the method is specifically directed for delivering a software program to a subscriber over a cable network which includes a head-end server. In operation, a plurality of software programs are compressed **39** at the head-end server and transmitted **41** over the cable network at corresponding predetermined scheduled times. A primary processor comprising, for example, a customized low end PC and hard disk is provided **43** in communication with the cable network. A secondary processor comprising, for example, a game machine which includes a game CPU and CD-ROM drive is provided **45** in communication with the primary processor. Operating instructions are generated **47** at the secondary processor for receipt by the primary processor so as to enable the primary processor to select and receive one of the plurality of software programs from the cable network without upstream communication to the head-end server. The selected software program is thereafter received **49** at the primary processor at the corresponding predetermined scheduled time whereupon it is decompressed **51** and temporarily stored **53**.

As discussed above, in a preferred embodiment, the method further comprises transmitting **55** broadcast data at

the head-end server for receipt by the primary processor. The broadcast data corresponds to a predetermined scheduled time that the software program is available to be received. Means is further provided at the secondary processor to convert **57** the broadcast data to graphic data for display on a television or monitor. In keeping with the invention, the transmitted software programs are decompressed at the primary processor on a data block basis or on the fly.

The User Interface

Turning now to FIG. **2**, there is shown a flow diagram of the user interface of the system of FIG. **1**. First, as indicated at block **40**, a Logo Screen appears on the display **36** after the primary processor **18** is turned on or when the subscriber has the network interface software **38** inserted into the secondary processor **20**, even if the primary processor **18** is not turned on.

The Logo Screen transitions to a Main Menu Screen after a predetermined amount of time, e.g., five seconds, or immediately if the subscriber presses a predetermined button on the game controller of the video game player **20**, as shown at block **42**. The subscriber may make selections using a video game controller coupled to the video game player, such as a mouse, game pad, joystick or keypad. The display **36** displays "buttons" on-screen which are selectable when highlighted. The subscriber preferably has at least three options available at the Main Menu. The subscriber may return to the Logo Screen, select a new game or restart the last game already loaded into the hard disk in the primary processor **18**.

If the subscriber selects a new game, the subscriber is presented with a "Select a Game" Screen, as shown at block **44**. The "Select a Game" Screen, as shown in FIG. **3**, provides the subscriber with at least two options. The subscriber may return to the Main Menu, as shown at block **46**, or select a game. The "Select a Game" Screen displays **24** hours worth of listings available to the subscriber. The grid is navigated by scrolling a highlight with the video game controller. The highlighted option may be selected by pressing a predetermined button such as "A" or "Play/Pause" on a video game controller. As shown in FIG. **3**, games are offered at 30 minute intervals with three time slots per screen. This interval may, of course, be varied to accommodate user preferences.

As the first listed time slot elapses, the elapsed time slot slides off the screen and a new set of time slots are displayed. The subscriber may thus download any one of the available games during the present time slot. The subscriber may also select a video game to be downloaded at a future time by selecting the appropriate game.

Once the subscriber selects a video game, a "Game Title" Screen appears, as shown at block **48**. The "Game Title" Screen displays a description of the game. The subscriber has at least two options. The subscriber may select Stop and return to the Main Menu, as shown at block **50**, or confirm the selection and advance to an "Ad/Countdown" Screen, as shown at block **52**. The "Ad/Countdown" Screen plays an advertisement on the screen and provides a countdown to when the game will be ready to play.

The subscriber has at least three options at the "Ad/Countdown" Screen. The subscriber may select Stop and return to the Main Menu, as shown at block **54**, return to the "Select a Game" Screen, as shown at block **56**, or select game information and advance to a "Game Information" Screen, as shown at block **58**. The "Game Information" Screen displays various options for selection, such as game instructions, high scores, hints and tips, and lobby arcade. The lobby arcade option allows the subscriber to play one of several preselected games stored on the start up CD-ROM while waiting for his/her selected game to be downloaded to the hard disk. The preselected games may be games such as chess, checkers, etc. The subscriber may then select one of the options, return to the Main Menu, as shown at block **60**, or return to the "Ad/Countdown" Screen, as shown at block **62**.

Once the game has finished downloading and is ready to be played, a "Game Ready" Screen appears, as shown at block **64**. The subscriber may then return to the Main Menu, as shown at block **66**, or begin playing the game. If the game is a multi-household game, "Start Playing Game" and "Multiplayer Match" Screen appear, as shown at blocks **68** and **70**. In the case of a multi-household game, the game can be played against other players who have the present invention in their home. A remote server connected to the primary processor **18** via a two-way cable network or via the central office of the local telephone network will locate and connect an opponent. Either the subscriber or the interactive network may attempt to locate an opponent. The interactive network may attempt to locate an opponent based on one of several selected criteria, such as same geographic location as the subscriber or same skill level as the subscriber.

Returning to the Main Menu, if the subscriber chooses to restart the last game, the "Game Information" Screen appears, as shown at block **72**. The subscriber has the options of returning to the Main Menu or starting the game, as shown at blocks **74** and **76**, respectively. As described above, if the game is a multi-household game the "Multiplayer Match" Screen appears, as shown at block **78**.

The present invention may also allow the subscriber to order the game. If such an option is available, an "Offer to Buy Game" Screen will appear after the game ends. The "Offer to Buy Game" Screen lists the purchase price of the game, time for delivery and purchase options, such as credit card or smart card. The subscriber may return to the Main Menu or proceed to buy the game. Upon deciding to purchase the game, the subscriber advances to a "Confirmation of Order" Screen that allows the subscriber to confirm the order or return to the Main Menu. The order is communicated by the system via a two-way cable network or via the central office of the local telephone network to a remote server where the order is processed and the item is shipped. An optional "Billing Information" Screen may be displayed to the subscriber to indicate the current charge and a new balance, if any.

The advantages of the present invention are numerous. The present invention supports cost effective and convenient (for the subscriber and the network provider) distribution of large amounts of data. The present invention also uses limited bandwidth on analog cable thereby eliminating expensive cable network upgrades. Furthermore, the present invention allows the network provider to deliver software written for video game and personal computer systems with minimal changes to the software/title as used in a stand-alone environment, i.e., addition of a wrapper, encryption, etc. Still further, the present invention is compatible with various video game systems available for purchase.

While the best modes for carrying out the invention have been described in detail, those familiar with the art to which this invention relates will recognize various alternative designs and embodiments for practicing the invention as defined by the following claims.

What is claimed is:

1. A system for delivering a software program over a cable network to a subscriber, comprising:

a head-end server operative to transmit program guide information periodically and compress a plurality of

software programs and transmit the compressed software programs over the cable network;

a primary processor in communication with the cable network for receiving said program guide information and displaying said information in a descriptive format and for receiving a selected one of said plurality of software programs and decompressing and storing the selected software program; and

a secondary processor in communication with the primary processor, the secondary processor including navigational software and drivers to enable the primary processor to receive and display said program guide information and to select and receive the software program from the cable network without upstream communication to the head-end server.

* * * * *

US006002852A

United States Patent [19]

Birdwell et al.

[11] Patent Number: 6,002,852

[45] Date of Patent: *Dec. 14, 1999

[54] **METHOD AND SYSTEM FOR CONFIRMING RECEIPT OF DATA OPPORTUNISTICALLY BROADCAST TO CLIENT COMPUTER SYSTEMS**

[75] Inventors: **Kenneth J. Birdwell**, Bellevue; **Steven J. Fluegel**, Redmond; **Patrick C. O'Hanlon**, Kirkland, all of Wash.

[73] Assignee: **Microsoft Corporation**, Redmond, Wash.

[*] Notice: This patent is subject to a terminal disclaimer.

[21] Appl. No.: **09/107,621**

[22] Filed: **Jun. 30, 1998**

Related U.S. Application Data

[63] Continuation of application No. 08/502,706, Jul. 14, 1995, Pat. No. 5,793,973.

[51] **Int. Cl.**[6] **G06F 13/00**

[52] **U.S. Cl.** **395/200.33**; 395/200.57; 395/200.62

[58] **Field of Search** 395/200.33, 200.62, 395/200.6, 200.67, 200.57; 370/312, 278

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,507,781	3/1985	Alvarez, III et al.	370/266
4,845,658	7/1989	Gifford	395/200.47
4,888,727	12/1989	Getson, Jr. et al.	395/876
4,958,278	9/1990	Meguro	395/200.49
5,459,725	10/1995	Bodner et al.	370/390
5,467,341	11/1995	Matsukane et al.	370/253
5,519,704	5/1996	Farinacci et al.	370/402
5,572,678	11/1996	Homma et al.	395/200.57
5,706,435	1/1998	Barbara et al.	711/141
5,793,973	8/1998	Birdwell et al.	395/200.53

OTHER PUBLICATIONS

S. Deering, "Host Extensions for IP Multicasting," *Stanford University, Network Working Group, Request for Comments*: 1112, Obsoletes: RFCs 988, 1054, Aug. 1989, pp. 1–21.

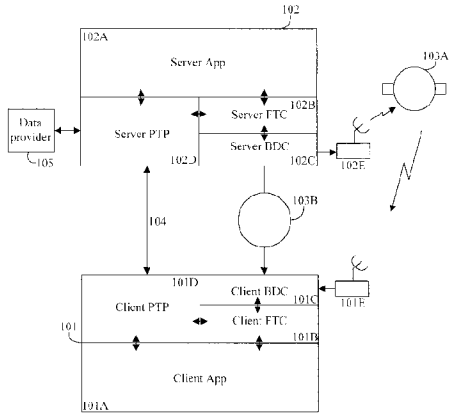
Primary Examiner—Krisna Lim

Attorney, Agent, or Firm—Seed and Berry LLP

[57] **ABSTRACT**

A method and system for opportunistically downloading data from a server computer system to client computer systems. The server computer system has a point-to-point transmission mechanism for receiving data from each client computer system and has a broadcast transmission mechanism for broadcasting data to the client computer systems. Each client computer system has a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism when the client computer system is in a receiving state. In a preferred embodiment, the server computer system selects data to be downloaded from the server computer system to the client computer systems, and broadcasts the selected data using the broadcast transmission mechanism. Each client computer system that is in the receiving state receives the broadcasted data and sends a confirmation that the client computer system has received the broadcasted data to the server computer system using the point-to-point transmission mechanism. Conversely, when a client computer system enters the receiving state, it sends a request to send the selected data to the server computer system using the point-to-point transmission mechanism. The server computer system receives the sent request and transmits the selected data to the client computer system that sent the request when the server computer system has not received confirmation that the client computer system that sent the request received the broadcasted data.

26 Claims, 14 Drawing Sheets



Legend	
APP	Application
BDC	Broadcast Data Componen:
FTC	File Transfer Component
PTP	Point-to-Point Connection

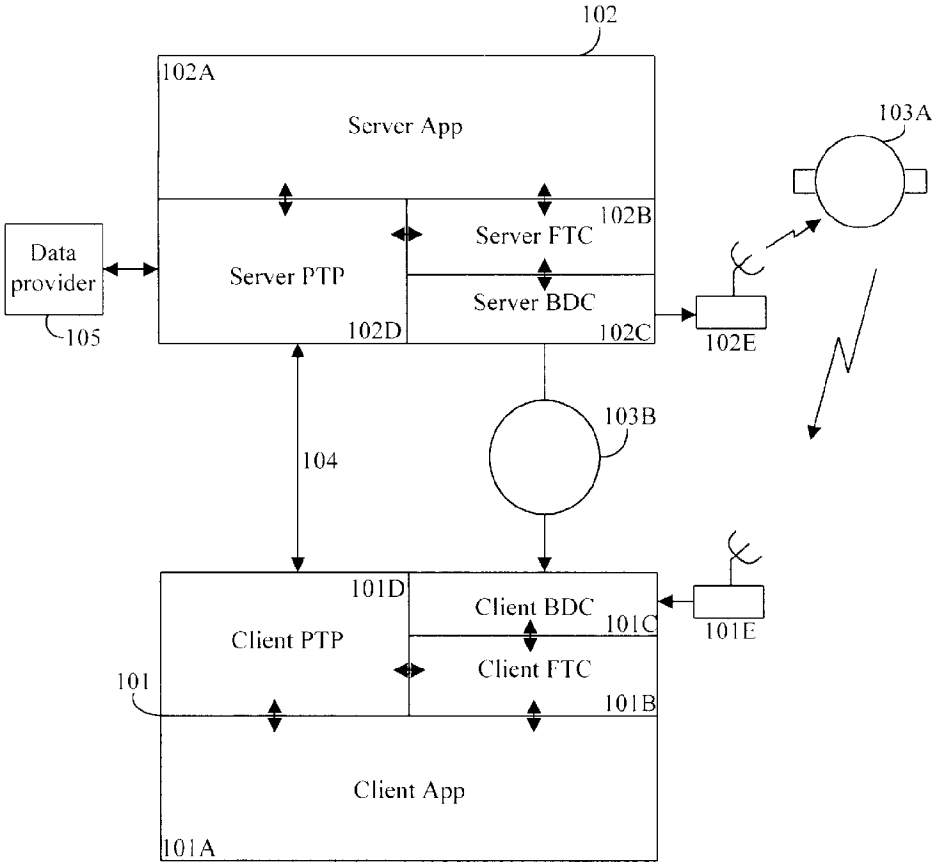


Fig. 1

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

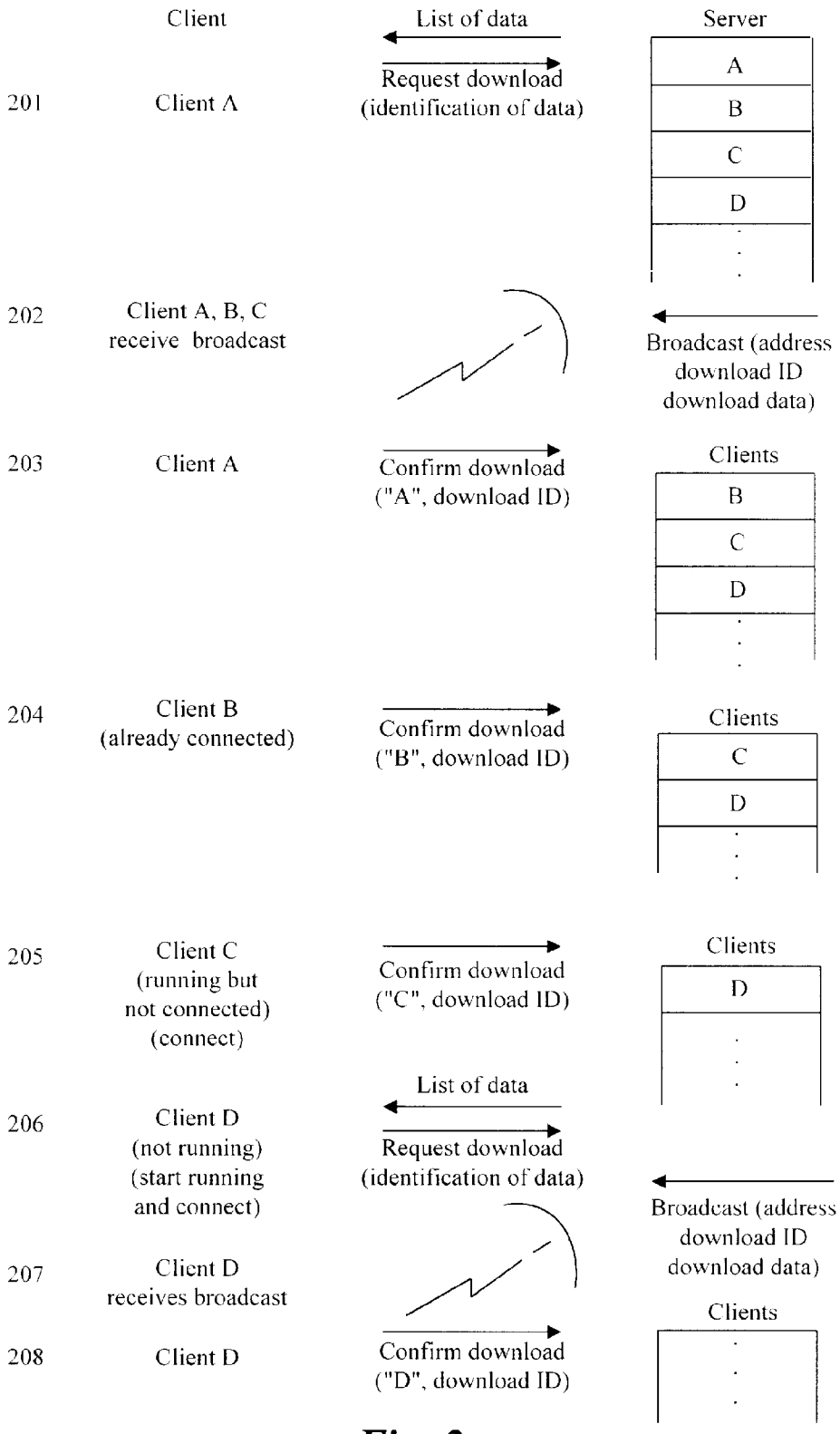

Client
List of data
Server
201
Client A
Request download
(identification of data)
A
B
C
D
202
Client A, B, C
receive broadcast
Broadcast (address
download ID
download data)
203
Client A
Confirm download
("A", download ID)
Clients
B
C
D
204
Client B
(already connected)
Confirm download
("B", download ID)
Clients
C
D
205
Client C
(running but
not connected)
(connect)
Confirm download
("C", download ID)
Clients
D
List of data
206
Client D
(not running)
(start running
and connect)
Request download
(identification of data)
Broadcast (address
download ID
download data)
207
Client D
receives broadcast
Clients
208
Client D
Confirm download
("D", download ID)

Fig. 2

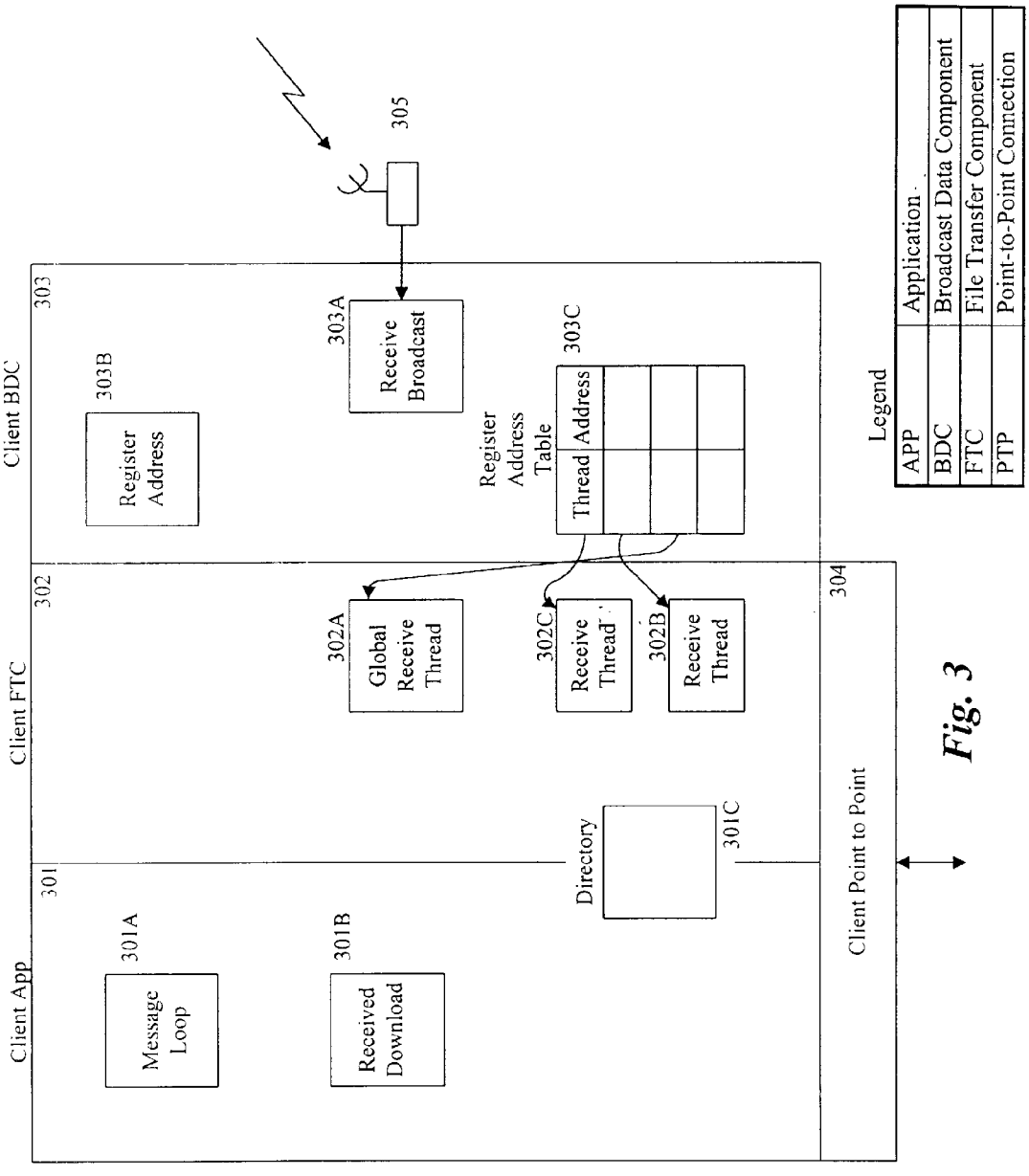

Client App
301
301A
Message Loop
301B
Received Download
Client FTC
302
302A
Global Receive Thread
302C
Receive Thread
302B
Receive Thread
Client BDC
303
303B
Register Address
303A
Receive Broadcast
Register Address Table
303C
Thread
Address
Directory
301C
Client Point to Point
304
305
Legend
APP
Application
BDC
Broadcast Data Component
FTC
File Transfer Component
PTP
Point-to-Point Connection

Fig. 3

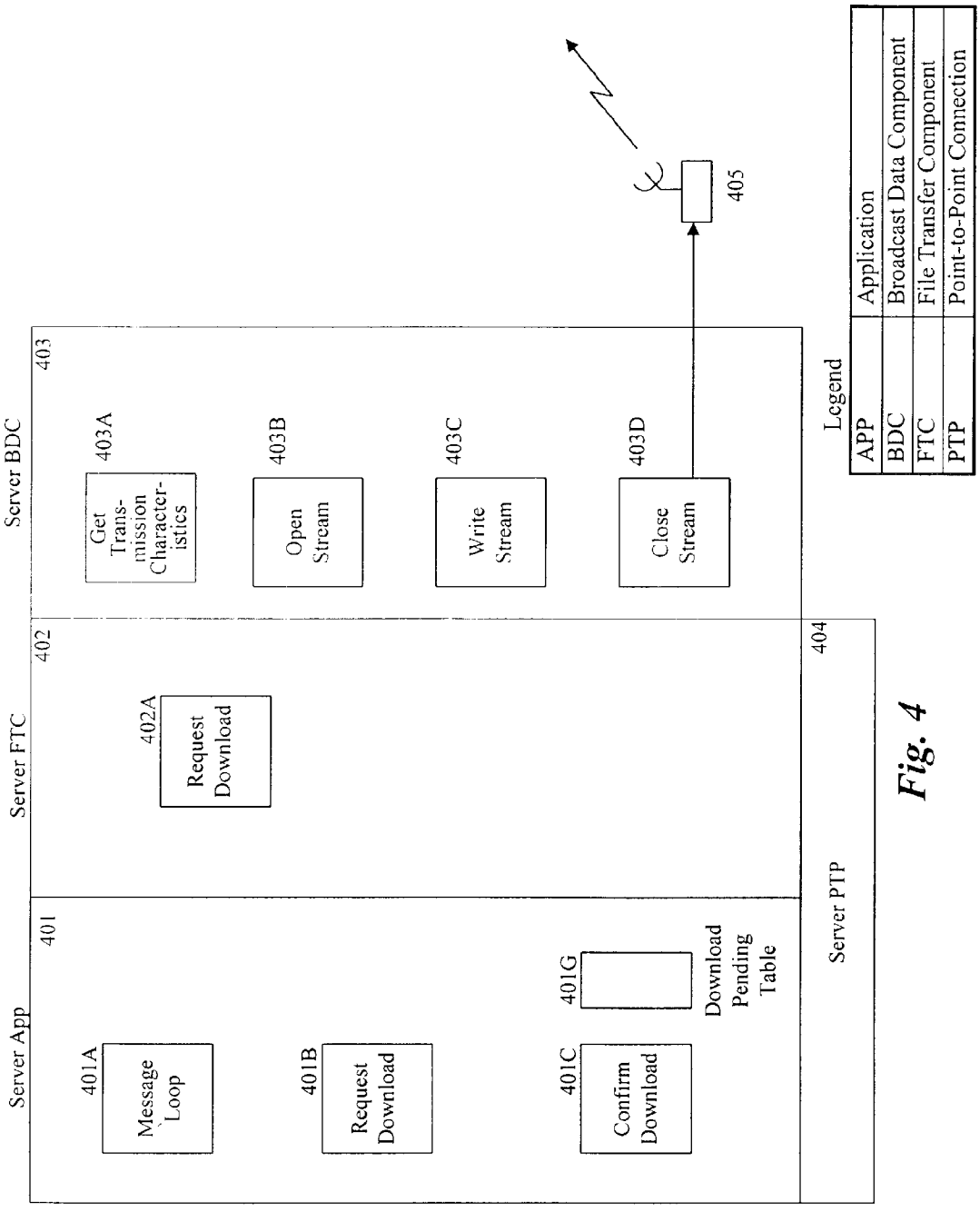

Server App
401
401A
Message Loop
401B
Request Download
401C
Confirm Download
401G
Download Pending Table
Server FTC
402
402A
Request Download
Server BDC
403
403A
Get Trans-mission Character-istics
403B
Open Stream
403C
Write Stream
403D
Close Stream
Server PTP
404
405
Legend
APP
Application
BDC
Broadcast Data Component
FTC
File Transfer Component
PTP
Point-to-Point Connection

Fig. 4

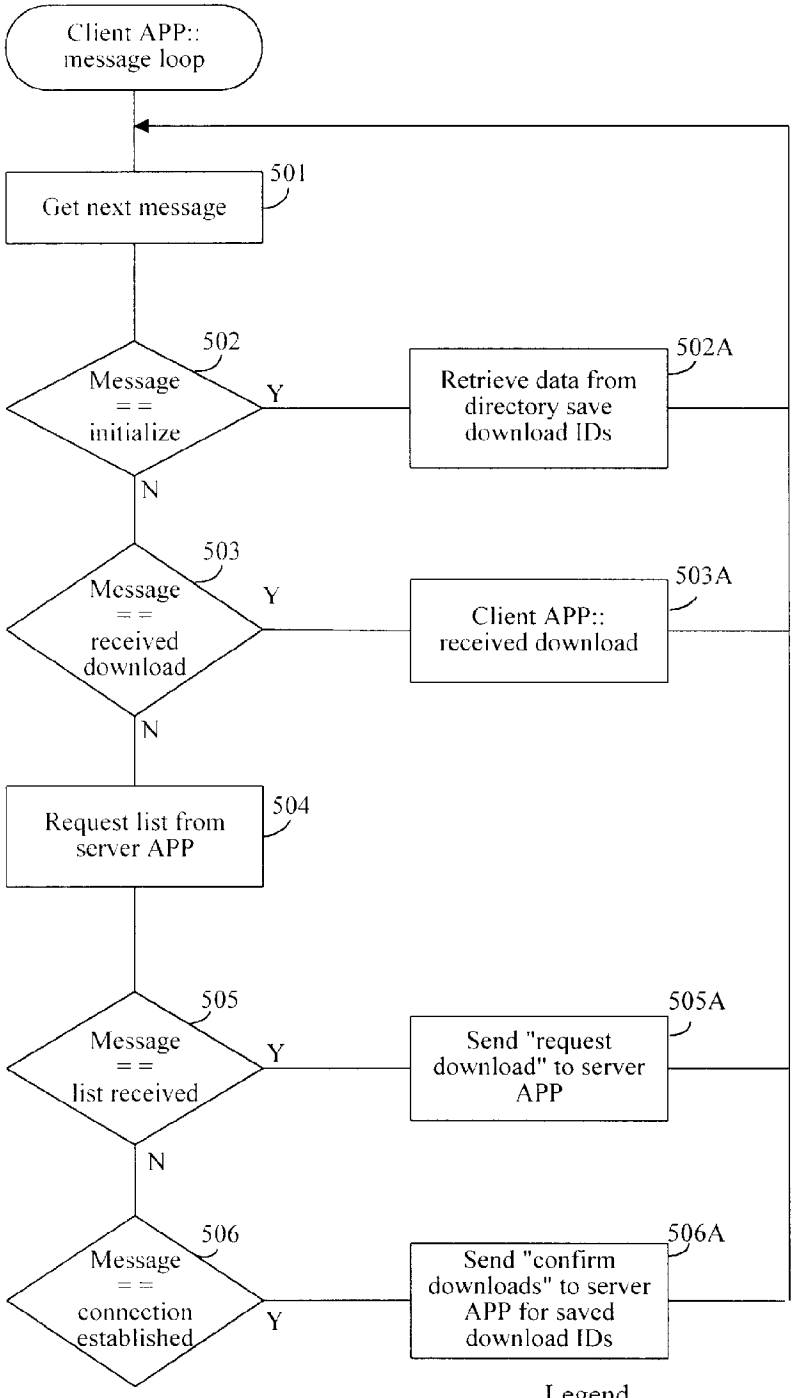


Fig. 5

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

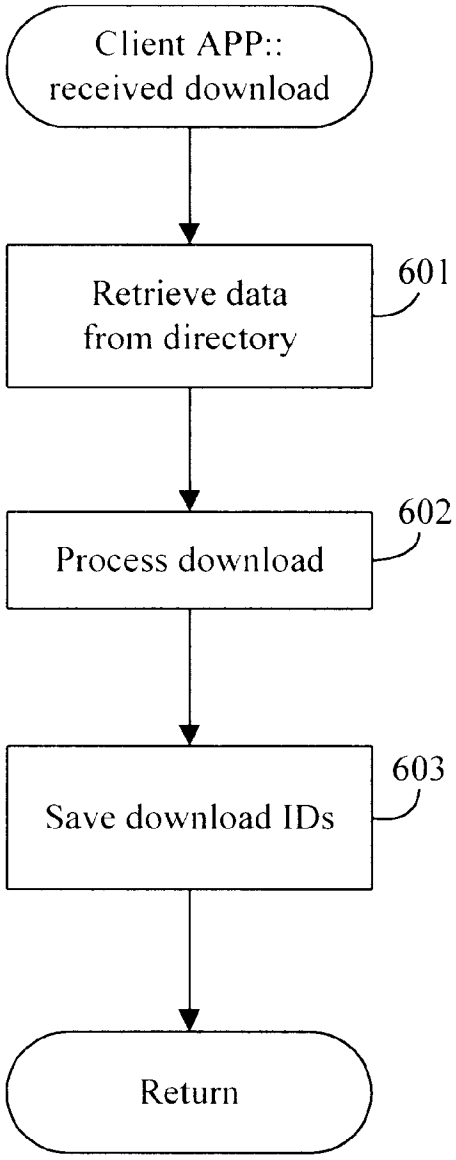

Client APP::
received download
Retrieve data
from directory
601
Process download
602
Save download IDs
603
Return

Fig. 6

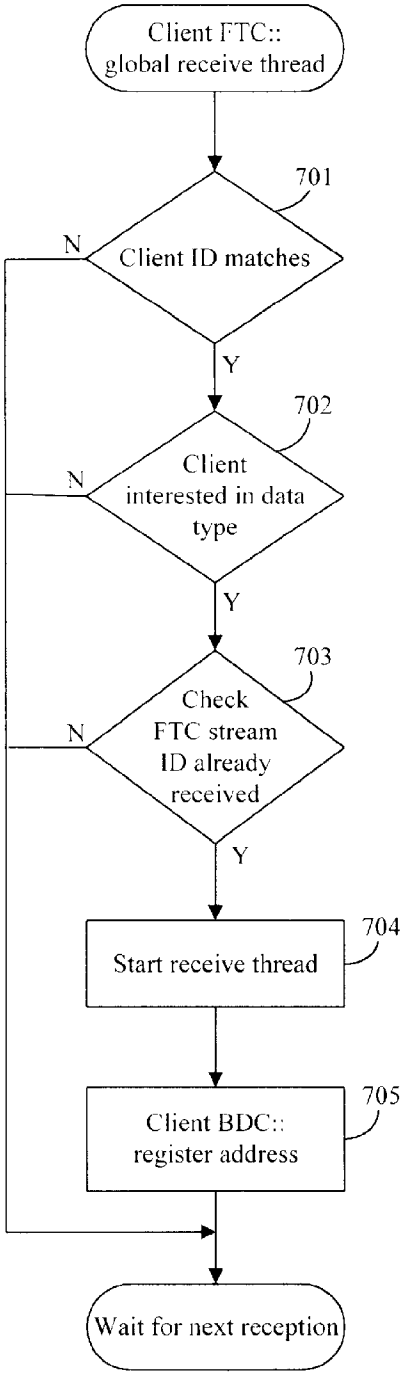


Fig. 7

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

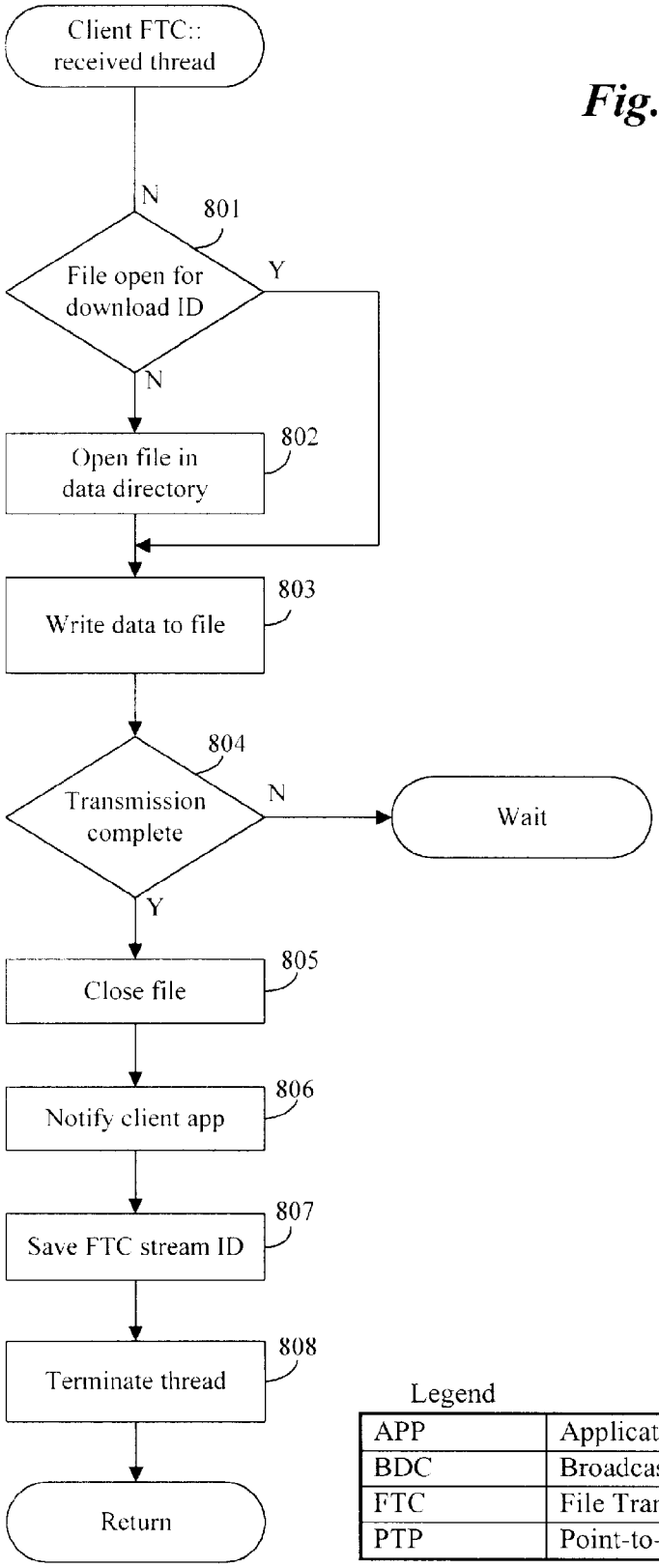


Fig. 8

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

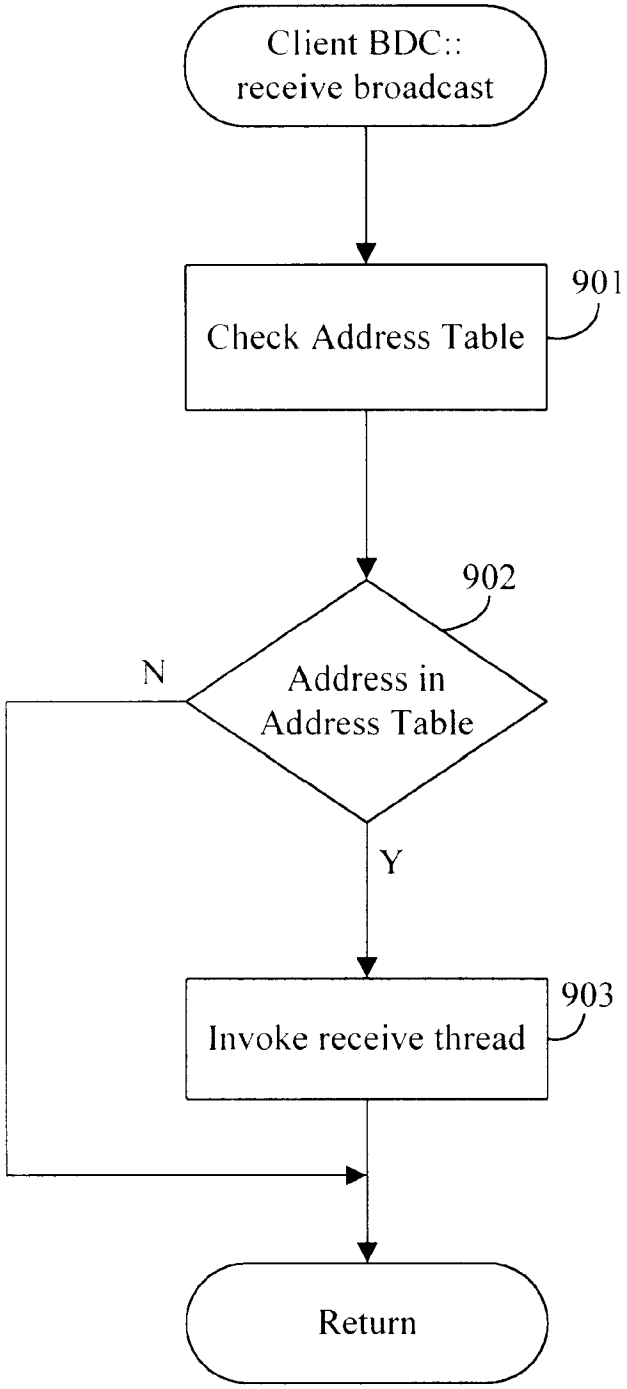

Client BDC::
receive broadcast
Check Address Table
901
902
Address in
Address Table
N
Y
Invoke receive thread
903
Return

Fig. 9

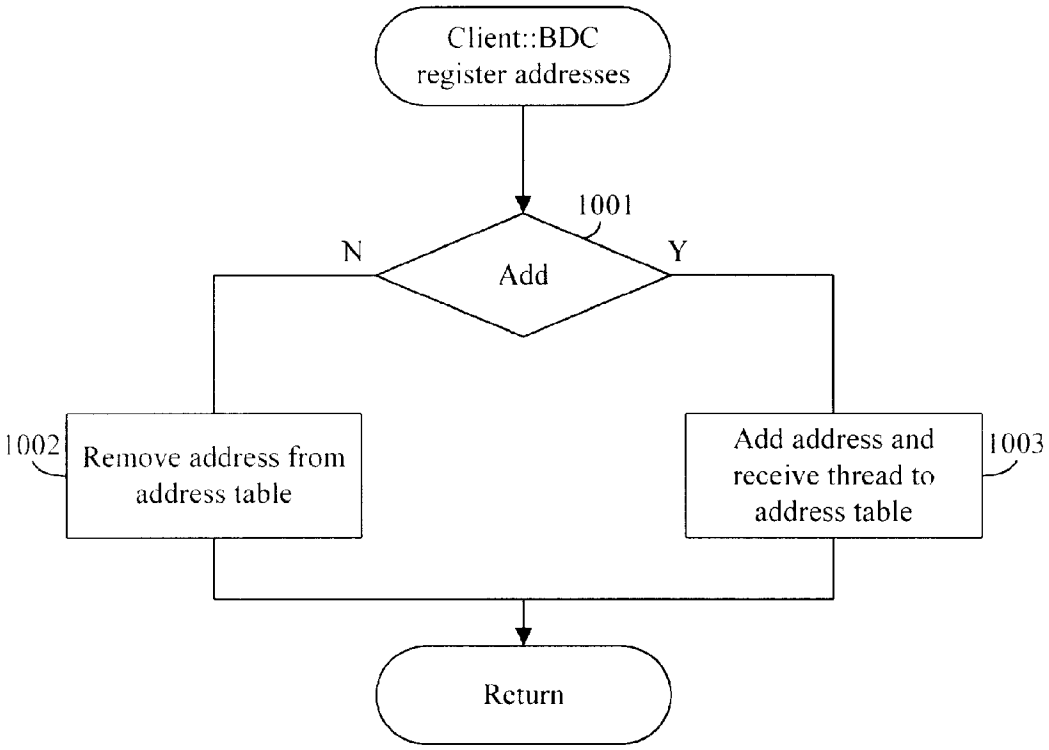

Client::BDC
register addresses
1001
Add
N
Y
1002
Remove address from
address table
Add address and
receive thread to
address table
1003
Return

Fig. 10

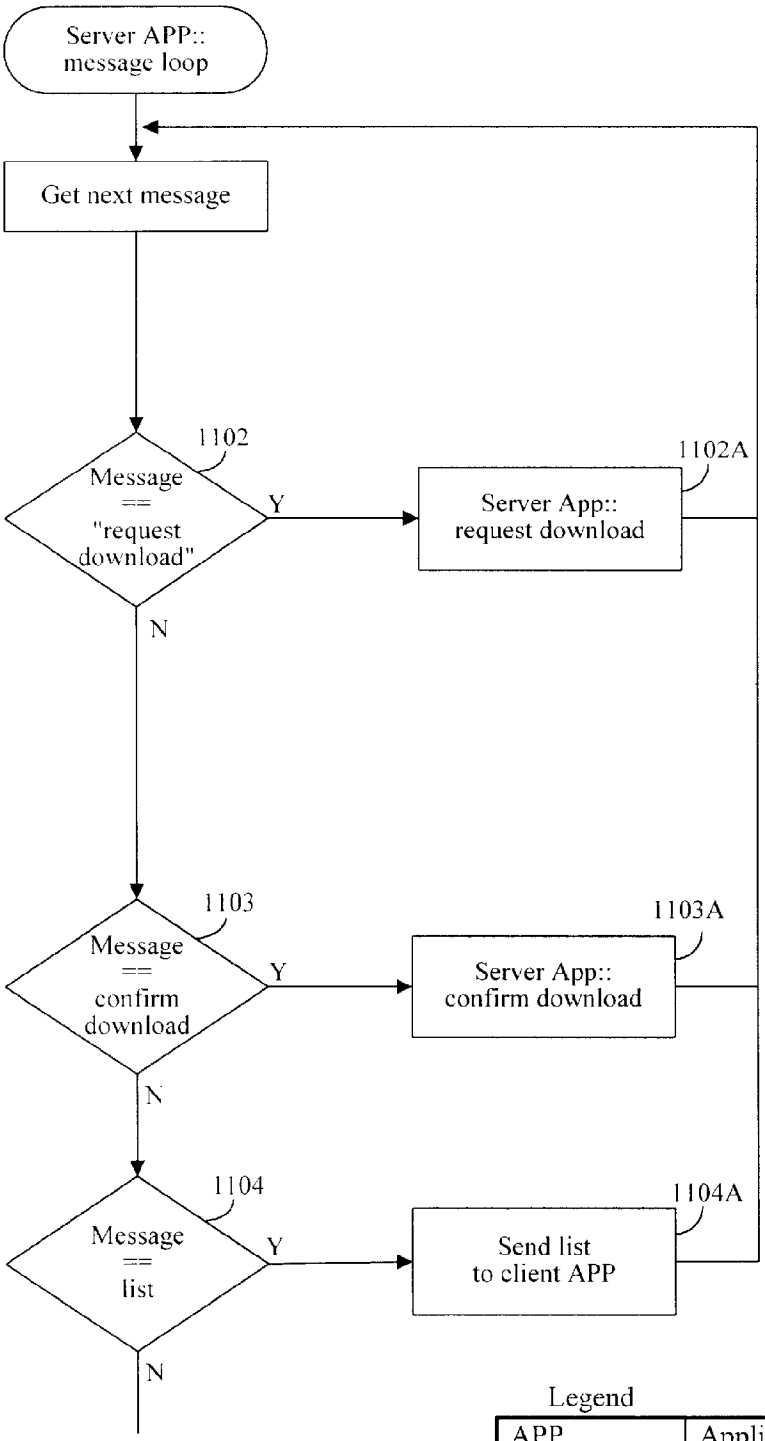


Fig. 11

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

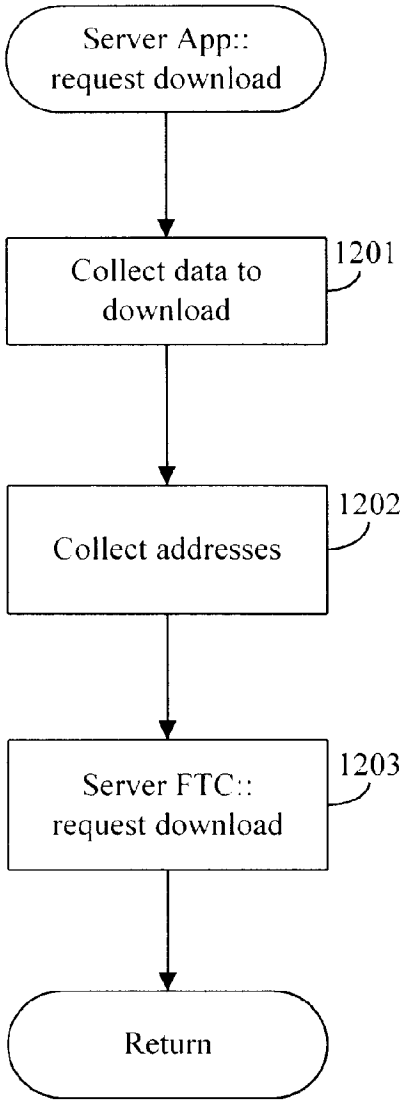


Fig. 12

Legend

APP	Application
BDC	Broadcast Data Component
FTC	File Transfer Component
PTP	Point-to-Point Connection

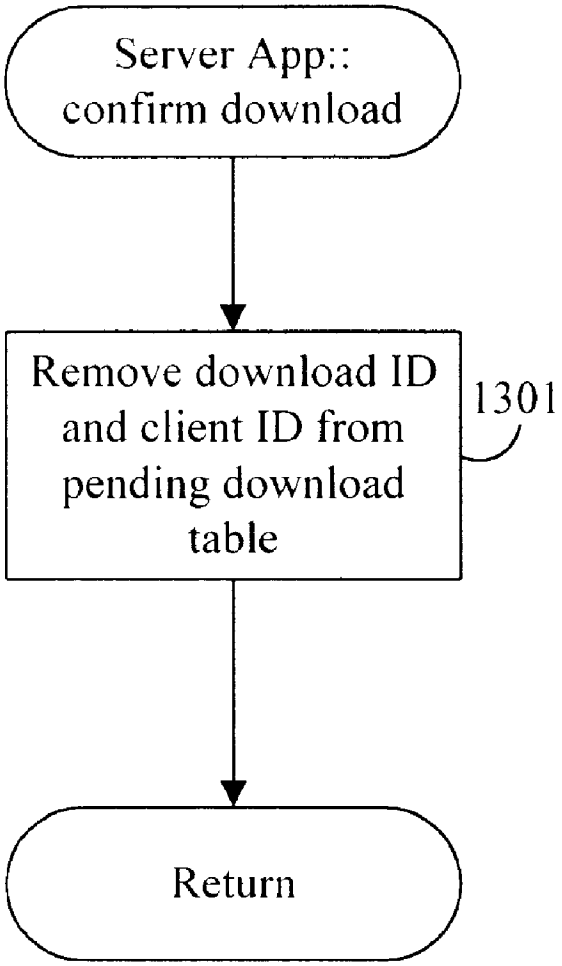

Server App::
confirm download
Remove download ID and client ID from pending download table
1301
Return

Fig. 13

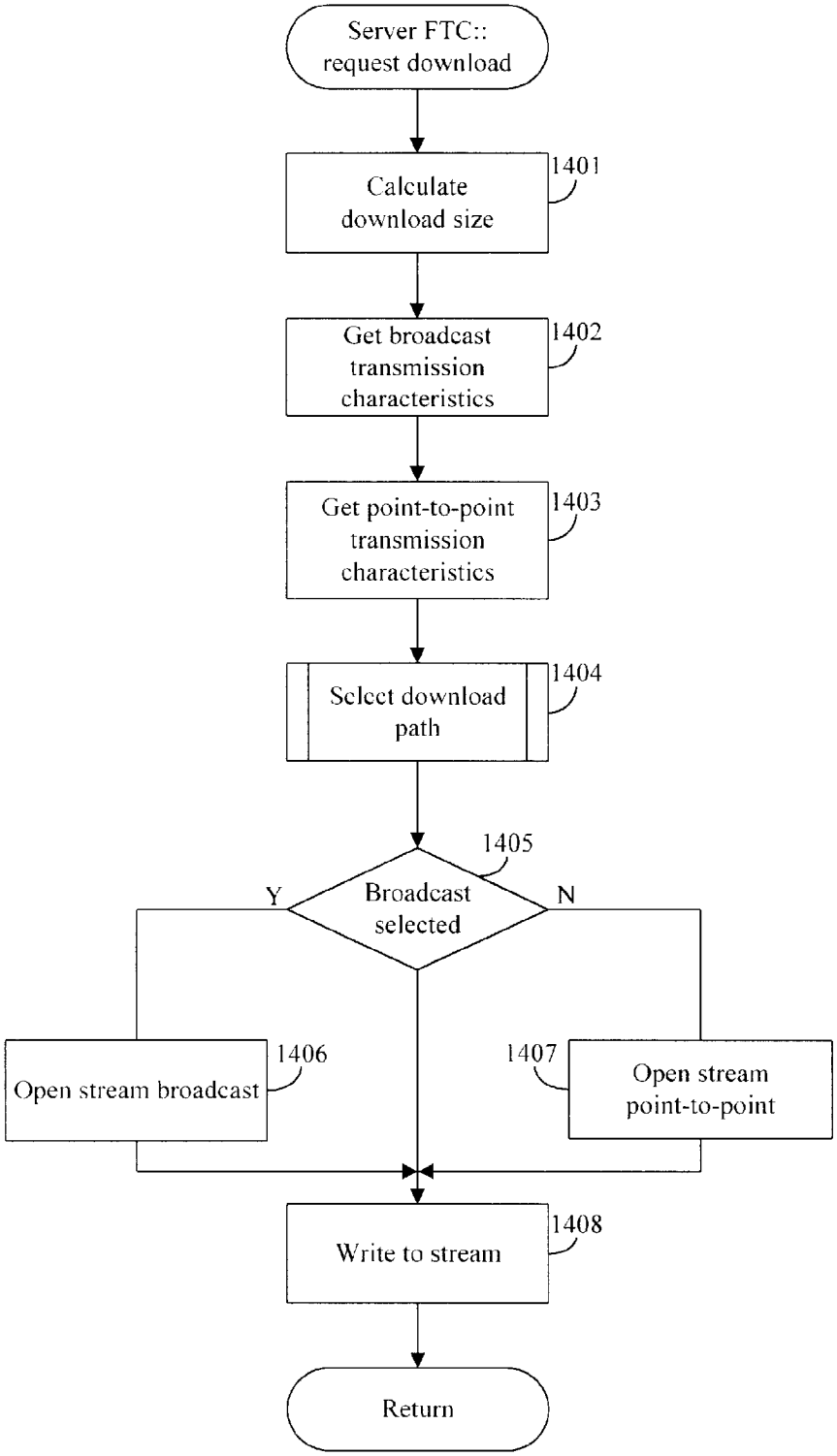

Server FTC:: request download
Calculate download size
1401
Get broadcast transmission characteristics
1402
Get point-to-point transmission characteristics
1403
Select download path
1404
1405
Broadcast selected
Y
N
Open stream broadcast
1406
1407
Open stream point-to-point
Write to stream
1408
Return

Fig. 14

METHOD AND SYSTEM FOR CONFIRMING RECEIPT OF DATA OPPORTUNISTICALLY BROADCAST TO CLIENT COMPUTER SYSTEMS

This application is a continuation of U.S. patent application Ser. No. 08/502,706, filed Jul. 14, 1995, now issued as U.S. Pat. No. 5,793,973.

TECHNICAL FIELD

This invention relates generally to a method and system for transmission of data and, more specifically, to a method and system of opportunistically broadcasting data or sending the data by a point-to-point connection.

BACKGROUND OF THE INVENTION

As computer systems become more and more common in the workplace and in the home, the demand for transmission of information between these computer systems is increasing. Many computer system are interconnected through networks such as the Internet. Although each computer system connected to such a network can receive data from and send data to each other computer system, there are difficulties in using such networks. For example, because such computer systems are based on the telephone network, data transmission typically occurs at a relatively slow transmission rate. Also, when the same data is to be sent to multiple computer systems, the same data must be transmitted multiple times, that is, once for each computer system that is to receive the data.

When many computer systems need to receive the same data, some computer systems broadcast the data using satellite transmission. With satellite transmission, the same data can be sent only once and received by many computer systems. Although satellite transmission rates are very fast, there are several disadvantages of transmitting data by satellite. First, the cost of transmitting data by satellite can be prohibitive. Second, the cost of each computer system is increased because each computer system needs a satellite receiver. Third, if a computer system is not running at the time of transmission, the computer system cannot receive the transmission. Consequently, satellite transmissions are often repeated at periodic intervals to ensure that each computer system receives the transmission. Of course, the repeating of such transmissions can be expensive and may be unnecessary because all computer systems, or least all those that care about the transmission, may have been running and actually received the data when it was first transmitted. Moreover, since such satellite transmissions typically only occur in one direction, there is generally no way of ensuring that all the intended recipient computer systems actually received a satellite transmission.

It would be desirable to have a transmission mechanism that would combine the advantages of the network computer systems and of satellite transmission, while minimizing their disadvantages.

SUMMARY OF THE INVENTION

The present invention provides method and system for opportunistically downloading data from a server computer system to client computer systems. The server computer system has a point-to-point transmission mechanism for receiving data from each client computer system and has a broadcast transmission mechanism for broadcasting data to the client computer systems. Each client computer system has a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism when the client computer system is in a receiving state. In a preferred embodiment, the server computer system selects data to be downloaded from the server computer system to the client computer systems, and broadcasts the selected data using the broadcast transmission mechanism. Each client computer system that is in the receiving state receives the broadcasted data and sends a confirmation that the client computer system has received the broadcasted data to the server computer system using the point-to-point transmission mechanism. Conversely, when a client computer system enters the receiving state, it sends a request to send the selected data to the server computer system using the point-to-point transmission mechanism. The server computer system receives the sent request and transmits the selected data to the client computer system that sent the request when the server computer system has not received confirmation that the client computer system that sent the request received the broadcasted data.

Another aspect of the present invention provides a method and system for ensuring receipt of data that is broadcast from a server computer system and received by client computer systems. The server computer system has a broadcast transmission mechanism. Each client computer system has a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism. The server computer system broadcasts the data using the broadcast transmission mechanism. Each client computer system uses the broadcast receiver to receive the broadcast data. Upon receiving the broadcasted data, each of client computer systems sends a confirmation from the client computer system to the server computer system through a transmission mechanism for sending data from client computer system to the server computer system.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram illustrating the components of the opportunistic broadcast system.

FIG. **2** is a diagram illustrating the operation of the opportunistic broadcasting system.

FIG. **3** is a block diagram illustrating the client APP, the client FTC and the client BDC.

FIG. **4** is a block diagram illustrating the server APP, server FTC, and the server BDC.

FIG. **5** is a flow diagram of the Message Loop of the client APP.

FIG. **6** is a flow diagram of the Received Download routine of the client FTC.

FIG. **7** is a flow diagram of the Global Receive thread of the client FTC.

FIG. **8** is a flow diagram of a sample Receive thread for the client FTC.

FIG. **9** is a flow diagram of the Receive Broadcast routine of the client BDC.

FIG. **10** is a flow diagram of the Register Address routine of the client BDC.

FIG. **11** is a flow diagram of the Message Loop of the server APP.

FIG. **12** is a flow diagram of the Request Download routine of the server APP.

FIG. **13** is a flow diagram of the Confirm Download routine of the server APP.

FIG. **14** is a flow diagram of the Request Download routine of the server FTC.

DETAILED DESCRIPTION OF THE INVENTION

The present invention provides a method and system for opportunistic broadcasting of data that is to be downloaded from a server computer system to client computer systems. In a preferred embodiment, a server computer system maintains a collection of data and can be connected to each client computer system through a point-to-point connection. The point-to-point connection can be via direct lines from each client computer system to the server computer system or via a routing network. Using the point-to-point connection, the server computer system can send data to and receive data from the client computer systems. However, if the same data is to be sent to multiple client computer systems using the point-to-point connection, the server computer system would send the same data multiple times: once for each client computer system. The server computer system also has a broadcasting transmission mechanism, such as a satellite, through which data can be broadcast to all client computer systems simultaneously. Each client computer system has a broadcast reception mechanism for receiving the broadcast data. Thus, when the same data is to be sent to multiple client computer systems, the server computer system can opportunistically broadcast the data and avoid sending the data once for each client computer system that is to receive the data. The server computer system sends the data via the broadcast transmission mechanism when it would be more efficient to do so.

In a preferred opportunistic broadcasting system of the present invention, the server computer system (server) receives a request from a client computer system (client) to download data from the server to the client computer system. The data is not only to be downloaded to the requesting client, but also to other clients who have not yet requested the data. The server may have received the data from a provider of data (e.g., contents of magazine) and a list of clients that are to receive the data. Alternatively, a client may have provided the data (e.g., electronic mail) that is to be sent to a list of clients. Also, the server may receive requests from many clients to download the same data. The server may group these multiple requests into a single download request that is to be downloaded to all the requesting clients. Before downloading the data to the requesting client, the server calculates certain transmission characteristics relating to the sending of the data to all the clients who are to receive the data using the point-to-point connection and using the broadcast mechanism. For example, the transmission characteristics may include transmission speed, cost of transmission, availability of the bandwidth for the transmission, and number of clients to whom the data is to be sent. The server uses these transmission characteristics to determine whether to transmit the data through the broadcast mechanism or through the point-to-point connection. For example, if the cost of transmitting through the broadcast mechanism is high and only two clients are to receive the data, then transmission through the point-to-point connection may be selected. Conversely, if the data is to be sent to 100 clients, then the cost of the 100 transmissions through the point-to-point connection may be higher than the cost of only one transmission through the broadcast mechanism. In this case, it may be cost-effective to broadcast the data. Thus, the opportunistic broadcasting system of the present invention selects the more efficient form of transmission based on the transmission characteristics.

When data is transmitted through the broadcast mechanism, not all the identified clients may be able to receive the broadcasted data. For example, a client computer system may be powered off at the time of the broadcast. The opportunistic broadcasting system uses a confirmation mechanism to ensure that each client eventually receives the data. The server tracks all those identified clients that have not yet confirmed receipt of the download data. When a client who has not confirmed receipt of the download data establishes a connection to the server through the point-to-point connection, the client may request the data to be downloaded. Typically, the server would send to the client a list of download data for which the client has not yet confirmed receipt. Using this list, the client can selectively request which data to download. To download the data again, the server again determines whether to transmit the data through point-to-point connection or through the broadcast mechanism. The server computer system makes this determination based on the transmission characteristics and the number of identified clients who have not yet confirmed receipt of the downloaded data. When the server re-transmits the download data for the clients who have not yet confirmed receipt, the server thus may again opportunistically broadcast the download data.

Many clients may be running at the time of broadcast, but may not be connected to the server at that time. These clients can receive and process the broadcasted data, but cannot confirm receipt of the broadcast. When such clients eventually connect to the server through the point-to-point connection, they can then confirm receipt of the download. Such clients would then typically request a list of data that the server has available to download to that client. However, since the client has confirmed receipt of some of the downloaded data, the list would not include that downloaded data. Thus, the client would have taken advantage of the broadcast of the data whose download was requested by another client.

FIG. **1** is a block diagram illustrating the components of the opportunistic broadcast system. The opportunistic broadcast system includes multiple client computer systems **101**, a server computer system **102**, broadcast transmission mechanisms **103**A, **103**B, point-to-point connection (PTP) **104**, and a data provider **105**. The client computer systems **101** include a client application (APP) **101**A, a client File Transfer Component (FTC) **101**B, a client Broadcast Data Component (BDC) **101**C, a client PTP component **101**D, and a broadcast receiver **101**E. The server computer system **102** includes a server APP **102**A, a server FTC **102**B, a server BDC **102**C, and a server PTP component **102**D, and a broadcast transmitter **102**E. The client computer systems can be connected to the server computer system through point-to-point connection **104**. To broadcast data, the server computer system transmits the data to the broadcast mechanism **103**A or **103**B. In a preferred embodiment, the broadcast mechanism **103**A is a satellite that receives transmissions from the broadcast transmitter **102**E and broadcasts the transmissions to all broadcast receivers **101**E simultaneously. Although the current preferred broadcast mechanism is satellite-based, the present invention can be used in conjunction with other types unidirectional broadcast mechanisms. For example, the computer systems may be connected to a cable television connection **103**B, a high-speed fiber optic channel, or a radio-frequency channel on which data can be broadcast to all client computer systems simultaneously. The data provider computer system **105** provides data to the server for transmission to identified clients. The client APP and server APP represent application programs that request and receive download data. For example, the application program may be an electronic mail system. The client APP may represent the client portion of

the mail system, and the server APP may represent the server portion of the mail system. The client APP and the server APP use the services of the client FTC and server FTC to download data. In a preferred embodiment, a client computer system can receive download data for a client APP even though the client APP is not currently executing and connected to the server APP. The client FTC and client BDC control the receiving of download data. If download data is received when the client APP is not executing, the data is stored until the client APP starts execution and retrieves the data.

Each transmission through the broadcast mechanisms includes information that identifies the client computer systems that are to receive the download data. Each client computer system is assigned a unique client identification number. The broadcast mechanism prefixes each transmission of download data with the client identifications of those client computer systems that are to receive the download data.

All client computer systems can receive all broadcasts of download data. However, only those client computer systems whose client identifications are included in the transmission will actually store and process the download data. The broadcast mechanism preferably uses an encryption mechanism to ensure that only the client computer systems to which the download data is being sent can store and process the download data. Each client computer system is assigned a public encryption key and a private decryption key. The server uses the public encryption key of the client to encrypt data that is to be sent to the client. Although all the clients can receive the encrypted data, only that client to which the data is directed can decrypt the data using its private decryption key.

The preferred broadcast mechanism uses "addresses," rather than client identification numbers when addressing a broadcast. Each client may be assigned its own address; in addition, a group of clients may be assigned to the same address. When a broadcast is addressed, each client that is assigned to that address can receive the download data. The broadcast mechanism need not be aware of the distinction between an address assigned to a single client or assigned to multiple clients. However, the server APP is preferably aware of each client that is assigned to an address. The server APP tracks whether all the clients assigned to an address have confirmed receipt of download data directed to the address. To ensure that the data sent to an address is received only by those clients who are assigned to that address, the server generates a public encryption key and a private decryption key for that address. Since each client assigned to an address needs access to the private decryption key to decrypt data sent to that address, the server sends the private decryption key to each assigned client. To send a private decryption key for an address to a client securely, the server encrypts the private decryption key using the public encryption key of the client before sending. Upon receiving the encrypted private decryption key of the address, the client uses its own private decryption key to decrypt the private decryption key of the address. In this way, only those clients that are assigned to an address have the private decryption key for that address. If new clients are assigned to the address, then the private encryption key for that address is sent to those new clients. If, however, clients are unassigned from the address, then a new public encryption key and private decryption key for the address is generated. The new private decryption key is sent to each client that is still assigned to the address. When data is sent to the address, it is encrypted with the new public encryption key.

In this way, the client that was unassigned from the address can no longer decrypt the data for that address.

In one embodiment, the preferred broadcast mechanism allows addresses to be dynamically assigned. To allow this dynamic assignment, each client is assigned to a global address. The server broadcasts new address assignments on the global address. The broadcast includes the new address and the client identifications of each client assigned to the new address. Since each client processes data for the global address, each client will know whether it is assigned to the new address. Those clients that are assigned to the new address can then register to receive data on the new address. The assigned clients would also be sent a private decryption key for the new address. Alternatively, each client that is assigned to the new address can individually be notified using the client's own address rather than the global address.

FIG. **2** is a diagram illustrating the operation of the opportunistic broadcasting system. In this example, a client computer system receives a list of available data that the server computer system is prepared to download to the client computer system. In response, the client requests that the data be downloaded from the server computer system. The data that client A is requesting is also to be sent to client B, client C, client D, etc. In this example, the server opportunistically broadcasts the data to client A so that the other clients may also receive the data. As shown in step **201**, client A sends a "request download" message to the server through the point-to-point connection. The message includes the identification of the data. When the server receives the "request download" message, the server recognizes that multiple clients are to receive the data and determines whether the data to be downloaded should be broadcast or sent through the point-to-point connection. In this example, the server decides to broadcast the data. The server first broadcasts to the global address that data will be transmitted on the identified address for clients A, B, C, D, etc. As shown in step **202**, the server then broadcasts the data on the identified address, and the data is received by clients A, B, and C who were assigned to that address. The server assigns a download identification for the broadcast, which is transmitted with the broadcast. The clients who receive the broadcast use this download identification to confirm receipt of the download data. At the time of the broadcast, clients A and B were connected to the server, but client C was not connected. The computer system of client C, however, was running and thus able to receive the broadcast. The computer system of client D was not running and thus was unable to receive the broadcast. As shown in step **202**, once client A receives the broadcast, it confirms receipt of the download by sending a "confirm download" message to the server. Upon receiving the "confirm download" message from client A, the server removes client A's identification from the list of clients who have not yet confirm the receipt of the download data. As shown in step **204**, client B, who also received the download data and is connected to the server, sends a "confirm download" message to the server. Upon receipt of the "confirm download" message from client B, the server removes client B's identification from the list of clients who have not yet confirmed receipt of the download. As shown in step **205**, client C was running, but was not connected to the server at the time of the broadcast. Although client C was not connected to the server, client C was able to process the data it received. When client C subsequently connects to the server, client C sends a "confirm download" message to the server. Upon receipt of the "confirm download" message, the server removes client C's identification from the list of clients who have not yet

confirmed receipt of the download data. This confirmation is referred to as a "delayed confirmation" because although client C received the download data, it delayed sending the "confirm download" message until it was connected to the server through the point-to-point connection. As shown in step **206**, client D eventually starts running and then connects to the server. Client D requests the server for a list of available data that is to be sent client D. The server determines that client D has not yet confirmed receipt of the download because its client identification is still in the list of clients. The server sends an indication to client D that it has this data for it. To download the data, client D then sends a "request download" message to the server. At that point, the server determines again whether the data to be downloaded should be broadcast or sent by the point-to-point connection to client D. In this example, the server decides to re-broadcast the data. As shown in step **207**, when the data is re-broadcast the server uses the global address to identify the address to which the data will be sent and the clients who are to receive the data, and then broadcasts the data using the same download identification. Client D then receives the broadcast. As shown in step **208**, client D sends a "confirm download" message to the server. The server removes client D from the list of clients. If a client had already received that download data, then it would disregard the re-broadcast. However, if a client had not received the initial broadcast, it would receive the re-broadcast, if running, and confirm receipt when it is connected to the server. Thus, the server can ensure that all the clients assigned to the address will eventually receive the download data or, at least, track those clients who have not yet confirmed receipt of the download data. Furthermore, the server can broadcast the data so that clients who have not yet requested the data to be downloaded can receive the data and avoid the overhead of requesting and retransmitting.

FIG. **3** is a block diagram illustrating the client APP, the client FTC, and the client BDC. The client APP represents a client application program that uses the services of the client FTC. The client APP receives a list of available data that may be downloaded, requests certain data be downloaded, receives the downloaded data, and confirms receipt of the downloaded data. During installation, the client APP registers the types of data that it would like to receive and designates the file directory in which it would like the data stored. The client FTC receives downloaded data, stores the data in the designated directory, and notifies the client APP that data has been received. The client BDC receives and decrypts the downloaded data for addresses that have been registered with it and notifies an appropriate thread of the client FTC. The client APP **301** contains a Message Loop **301**A, a Received Download routine **301**B, and a directory **301**C. The Message Loop **301**A receives messages sent to the client FTC and distributes the messages to the appropriate routines for handling. If broadcasts were received when the client APP was not connected to the server APP, then, upon connection, the client APP sends a "confirm download" (i.e., a delayed confirmation) message to the server APP for each download data received. The Received Download routine **301**B retrieves data from the directory, processes the data in the directory, and saves the download identifications for later confirmation to the server APP.

The client FTC contains Receive threads **302**A, **302**B, and **302**C. The Global Receive thread **302**A receives data for the global address. Certain data that is sent on the global address indicates that the client APP is to receive data on another address. When such data is received, the Global Receive thread **302**A starts another Receive thread **302**B and **302**C for receiving the data on that other address. These Receive threads are invoked by the client BDC whenever data for a download is received for an address that the client is assigned to and that the client is registered to receive. The Receive threads **302**B and **302**C store the data in a file in the directory designated by the client APP.

The client BDC **303** contains a Received Broadcast routine **303**A, a Register Address routine **303**B, and an Address table **303**C. The client FTC invokes the Register Address routine **303**B to register any addresses to which the client is assigned and for which the client wishes to receive broadcasts. The Receive thread associated with the address is invoked when broadcast data is received for the registered address. The Register Address routine stores the address and the identification of the Receive thread in the Address table. The Received Broadcast routine is invoked by a broadcast device driver to process any broadcasts that are received through broadcast receiver **305**. The Received Broadcast routine determines whether the broadcast address corresponds to any address in the Address table. If the Received Broadcast routine determines that there is a correspondence, then the Received Broadcast routine invokes the identified Receive thread to process the received data.

FIG. **4** is a block diagram illustrating the server APP, server FTC, and the server BDC. The server APP **401** contains a Message Loop **401**A, a Request Download routine **401**B, a Confirm Download routine **401**C, and a Download Pending table **401**G. The Message Loop **401**A receives each of the messages sent to the server APP from a client APP through the server PTP **404** and invokes the appropriate routine for handling the message. The Request Download routine **401**B processes "request download" messages. The Confirm Download routine **401**C processes "confirm download" messages. The Download Pending table **401**G identifies those downloads for which not all of the clients have yet confirmed receipt of the download data. The server APP adds an entry to this table for each client that is to receive the download.

The server FTC contains the Request Download routine **402**A for processing download requests from the server APP.

The server BDC **403** contains a Get Broadcast Transmission Characteristics routine **403**A, an Open Stream routine **403**B, a Write Stream routine **403**C, and a Close Stream routine **403**D. The Get Broadcast Transmission Characteristics routine retrieves various characteristics for the broadcast of the download data. The stream routines control the broadcasting of the download data through the broadcast transmission mechanism **405**.

Client Routines

FIGS. **5**–**10** are flow diagrams of the routines of the client computer system. Upon installation of the client APP, the client APP designates in which directory its downloaded data is to be stored and which types of data it wants downloaded. For example, an electronic mail application would want to download electronic mail type data, but not, for example, weather related data.

FIG. **5** is a flow diagram of the Message Loop of the client APP. The Message Loop receives messages from the server APP and the client FTC, confirms downloads, and processes download data. The client FTC receives the messages: "initialize," "received download," "list received," and "established connection." The "initialize" message indicates that the client APP has just started running. The "received download" message indicates that the client FTC has just received a download for the client APP. The "list received" message indicates that a list of data that may be downloaded

has been received from the server APP. The "established connection" message indicates that the client APP has established a connection with the server APP. Although many of the routines are described as using a message passing mechanism, one skilled in the art would appreciate that techniques of the present invention could also be implemented using a procedure call or remote procedure call mechanism. In step **501**, the Message Loop retrieves the next message. In steps **502**, if the message is "initialize," then the Message Loop continues at step **502**A, else the Message Loop continues at step **503**. In step **502**A, the Message Loop processes the data in the directory that was downloaded while the client APP was not running and saves the download identification of that data for later confirmation. In step **503**, if the message is "received download," then the Message Loop invokes the Received Download routine of the client APP in step **503**A. In step **504**, the Message Loop requests a list of the available data from the server APP by sending a "request list" message to the server APP. In step **505**, if the message is "list received," the Message Loop sends a "request download" message to the server APP for each data download it wants to receive in step **505**A. In step **506**, if the message is "established connection," then the Message Loop sends a "confirm download" message to the server APP for each download identification that was saved while the client APP was not connected to the server APP.

FIG. **6** is a flow diagram of the Received Download routine of the client APP. The Received Download routine retrieves the downloaded data from the directory, processes the data, and saves the download identification so that when a connection is established its receipt can be confirmed. In step **601**, the Received Download routine retrieves the file from the designated directory. In step **602**, the Received Download routine processes the retrieved file. In step **603**, the Received Download routine saves the download identification for later confirmation.

FIG. **7** is a flow diagram of the Global Receive thread of the client FTC. The Global Receive thread receives all the data sent on the global address. One type of global data indicates that data is to be transmitted to certain clients on another address. The global data includes what type of data will be transmitted, the identification of the clients to receive the data, and the identification of the address to which the data will be transmitted. If the client is identified, if the client APP has registered to receive that type of data, and if the client FTC has not already received the data, then the Global Receive thread sets up a mechanism for receiving the data on the identified address. In step **701**, if the client matches a identified client in the global data, then the thread continues at step **702**, else the thread disregards the data. In step **702**, if the client APP has register to receive that type of data, then the thread continues at step **703**, else the thread disregards the data. In step **703**, if the data has already been received, then the thread continues at step **704**, else the thread disregards the data. This happens, for example, when the downloaded data is re-broadcast because some addressee client computer systems have not confirmed receipt of the last broadcast. In steps **704**, the Global Receive thread starts a thread to receive the data to be transmitted. In step **705**, the Global Receive thread registers the new address with the client BDC.

FIG. **8** is a flow diagram of a sample Receive thread for the client FTC. The Receive threads handle the storing of streams of download data for a particular address. In step **802**, if a file has already been opened for the download data, then the thread continues at step **804**, else the thread opens a file in the designated directory for the client APP to receive the download data in step **803**. In step **804**, the Receive thread writes the received data to the open file. In step **805**, if the download has been completed, then the thread continues at step **804**, else the thread waits for the next transmission for that download. In step **806**, the Receive thread closes the file. In step **807**, the Receive thread sends a "received download" message to the client APP, if running, to notify the client APP that data has been downloaded for it. In step **808**, the Receive thread saves the identification of the data it received so that the Global Receive thread will disregard the data if it is re-broadcasted. In step **808**, the Receive thread removes its corresponding address from the Address table and terminates.

FIG. **9** is a flow diagram of the Receive Broadcast routine of the client BDC. The Receive Broadcast routine is invoked by the broadcast receiver device driver each time a broadcast is received. In step **901**, the Receive Broadcast routine checks the Address table to determine if the broadcast is directed to an address for which the client has registered to receive broadcasts. In step **902**, if the address of the broadcast is in the Address table, then the routine continues at step **903**, else the routine returns. In step **903**, the Receive Broadcast routine invokes the Receive thread identified in the Address table and returns.

FIG. **10** is a flow diagram of the Register Address routine of the client BDC. The Register Address routine is passed the address for which the client wants to receive data (or stop receiving data) and an identification of a Receive thread to handle the stream of broadcast data. In step **1001**, if the routine is invoked to add an address, then the routine continues at step **1003**, else the routine continues at step **1002**. In step **1002**, the Register Address routine removes the passed address from the Address table and returns. In step **1003**, the Register Address routine adds the passed address and the identification of Receive thread to the Address table and returns.

Server Routines

FIGS. **11**–**14** are flow diagrams of the routines of the server. FIG. **11** is a flow diagram of the Message Loop of the server APP. The server APP receives from the client APP three messages: "download request," "confirm download," and "request list." The "download request" message is a request to download data to the client APP. The "confirm download" message indicates that the client APP has received the data. The "request list" message indicates that the client APP wants to receive the list of available data that the server has yet to download to the client APP. The Message Loop calls the appropriate routine to handle the message. In step **1101**, the Message Loop retrieves the next message. In steps **1102**–**1104**, the Message Loop decodes the message. In steps **1102**A–**1104**A, the Message Loop processes the message. In step **1104**A, the server APP retrieves the list of available data to download for the client APP from the Download Pending table and sends the list to the client APP.

FIG. **12** is a flow diagram of the Request Download routine of the server APP. In step **1201**, the routine collects the data to be downloaded as indicated in the message received from the client APP. In step **1202**, the routine determines the clients to which the data is directed and that have not yet confirmed receipt as indicated by the Pending Download table. In step **1203**, the routine invokes the Request Download routine of the server FTC and returns.

FIG. **13** is a flow diagram of the Confirm Download routine of the server APP. In step **1301**, the routine removes the client identification from the Pending Download table

for the download identification for which the client confirmed receipt and returns.

FIG. **14** is a flow diagram of the Request Download routine of the server FTC. The Request Download routine determines which transmission path to use and transmits the download data. Alternatively, this routine could be passed an indication as to whether the data should be sent by the point-to-point or should be broadcasted. In step **1401**, the Request Download routine determines the size of the data to be downloaded. In step **1402**, the Request Download routine invokes a routine provided by the server BDC to determine the transmission characteristics for the broadcast. In step **1403**, the Request Download routine invokes a routine provided by the server point-to-point component to determine the transmission characteristics of the point-to-point transmission. In step **1404**, the Request Download routine evaluates the transmission characteristics to select either the broadcast transmission mechanism or point-to-point connection. In step **1405**, if broadcast mechanism is selected, then the routine continues at step **1406**, else the routine continues at step **1407**. In step **1406**, the Request Download routine opens a broadcast stream. In step **1407**, the Request Download routine opens a point-to-point stream. In step **1408**, the Request Download routine writes the download data to the open stream and returns.

In one embodiment of the present invention, certain transmission characteristics of a broadcast transmission are estimated based on bandwidth that was unused in a previous time interval. For example, if 10 K bytes per second of bandwidth was unused on average in the last 10 seconds and a request for transmission characteristics specifies an 1 M byte transmission, then the time of transmission is estimated as 100 seconds (i.e., 1 M bytes/10 K bytes per second). Other transmission characteristics may be based on cost. For example, a request for transmission may indicate that the transmission is to occur during non-prime time or during prime time. The transmission characteristics of the point-to-point connection can be determined in an analogous manner. For example, if the point-to-point connection is a 14 K baud phone line, then an 1 M byte transmission may take approximately 570 seconds (i.e., 1 M byte * 8 bits per byte/14 K bits per second). Similar costs would apply to prime and non-prime time point-to-point transmissions. Once the transmission characteristics are received, the server FTC determines the more efficient way to transmit the data. The server FTC would take into consideration the speed of transmission, cost of transmission, number of recipients, and the priority of the transmission.

Although the present invention has been described in terms of a preferred embodiment, it is not intended that the system be limited to this embodiment. One skilled in the art would appreciate that present invention can be used in conjunction With file servers, database server, or electronic mail servers. Thus, the download data can represent a file, a query from the database, or an electronic mail message. In addition, the present invention can be used for the distribution of computer programs and of real time data such as stock prices and weather information. Also, in an alternate embodiment, the server FTC, rather than the server APP, could track clients who have not yet confirmed receipt and periodically retransmit the data. Modifications within the spirit of the system will be apparent to those skilled in the art. The scope of the present invention is defined by the claims that follow.

We claim:

1. A method in a client computer system for confirming receipt of data broadcast by a server computer system, the broadcasted data being broadcasted using a broadcast transmission mechanism for transmitting data unidirectionally from the server computer system to the client computer systems, the method comprising:

when the client computer system does not have a connection established with the server computer system, receiving the broadcasted data; and

storing an indication that the broadcasted data has been received; and

when the client computer system later establishes a connection with the server computer system,

checking for presence of the stored indication that the broadcasted data has been received; and

when the stored indication is present, sending a confirmation that the client computer system received the broadcasted data to the server computer system using the established connection

so that the server computer system can track those client computer systems that have sent confirmations and can send the broadcasted data to those client computer systems that have not sent confirmations.

2. The method of claim **1** including after storing the indication, establishing a connection with the server computer system.

3. The method of claim **2** wherein the connection is established with a point-to-point transmission mechanism.

4. The method of claim **1** wherein when the stored indication is not present, sending an indication that the client computer system has not received any broadcasted data to the server computer system via the established connection.

5. The method of claim **1** wherein the server computer system resends the broadcasted data using the broadcast transmission mechanism.

6. The method of claim **1** wherein the server computer system resends the broadcasted data using the established connection.

7. A method for ensuring receipt of data that is broadcast from a server computer system and received by a plurality of client computer systems, the server computer system having a broadcast transmission mechanism for transmitting data unidirectionally from the server computer system to the client computer systems, each client computer system having an identity and a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism, the method comprising:

broadcasting the data using the broadcast transmission mechanism;

receiving at each of the plurality of client computer systems, the broadcasted data using the broadcast receiver;

upon receiving the broadcast data at a client computer system, sending a confirmation from the client computer system to the server computer system through a point-to-point transmission mechanism for sending data from client computer system to the server computer system; and

upon receiving by the server computer system a confirmation from a client computer system,

determining the identity of the client computer system that sent the confirmation; and

storing the identity of the client computer system that sent the confirmation so that the identity of a client computer system that did not receive the broadcasted data can be determined.

8. The method of claim **7** including:

upon receiving the broadcasted data at each of the plurality of client computer systems,

determining whether the client computer system has a connection established with the server computer system through the transmission mechanism; and

when it is determined that no connection has been established, delaying sending the confirmation until a connection is established.

9. A method in a server computer system for transmitting data from the server computer system to a plurality of client computer systems, the method comprising:

broadcasting the data using a unidirectional transmission mechanism so that at least some of the client computer systems receive the data; and

for each of the plurality of client computer systems, p**2** establishing a connection with the client computer system using a bidirectional transmission mechanism;

determining whether the client computer system has confirmed receipt of the broadcasted data; and

when it is determined that the client computer has not confirmed receipt of the broadcasted data, transmitting the data using the bidirectional transmission mechanism to the client computer system.

10. The method of claim **9** wherein when a client computer system receives the broadcasted data, the client computer system sends a confirmation of receipt to the server computer system via the bidirectional transmission mechanism.

11. The method of claim **10** wherein the server computer system determines that a client computer system has not confirmed receipt of the broadcasted data, when the server computer system has not received a confirmation from that client computer system.

12. A computer-readable medium containing instructions for causing a computer system to ensure receipt of data that is broadcast from a server computer system and received by a plurality of client computer systems, the server computer system having a broadcast transmission mechanism for transmitting data unidirectionally from the server computer system to the client computer systems, each client computer system having a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism, by:

broadcasting the data using the broadcast transmission mechanism;

receiving at each of the plurality of client computer systems, the broadcasted data using the broadcast receiver; and

upon receiving the broadcasted data at each of the plurality of client computer systems, sending a confirmation from the client computer system to the server computer system through a bidirectional transmission mechanism for sending data between client computer system and the server computer system

whereby the server computer system can track the client computer systems that have received the broadcasted data based on the sent confirmations.

13. The computer-readable medium of claim **12** including:

upon receiving the broadcasted data at each of the plurality of client computer systems,

determining whether the client computer system has a connection established with the server computer system through the bidirectional transmission mechanism; and

when it is determined that no connection has been established, delaying sending the confirmation until a connection is established.

14. A method for downloading data from a server computer system to a plurality of client computer systems, the server computer system having a broadcast transmission mechanism and a point-to-point transmission mechanism, the broadcast transmission mechanism for transmitting data from the server computer system to the client computer system but cannot transmit data from the client computer systems to the server computer system, each client computer system having a broadcast receiver for receiving data broadcast by the broadcast transmission mechanism, the point-to-point transmission mechanism being for transmitting data to and receiving data from each of the plurality of client computer systems, the method comprising:

determining broadcast transmission characteristics for the transmission of the data to be downloaded;

determining point-to-point transmission characteristics for the transmission of download data;

selecting whether to transmit the download data using the broadcast transmission mechanism or using the point-to-point transmission mechanism based on the determined transmission characteristics;

when the broadcast transmission mechanism is selected, broadcasting the data to be downloaded using the broadcast transmission mechanism; and

when the point-to-point transmission mechanism is selected,

sending the data to be downloaded to each client computer system using the point-to-point transmission mechanism.

15. The method of claim **14** including:

receiving confirmations from client computer systems that received the broadcasted data; and

when confirmations have not been received from all the client computer system, transmitting the data to be downloaded to those client computer systems for which a confirmation has not been received using the broadcast transmission mechanism or using the point-to-point transmission mechanism.

16. The method of claim **15** wherein the transmitting of the data to be downloaded to those client computer systems for which a confirmation has not been received is periodically repeated.

17. The method of claim **15** wherein the step of transmitting is periodically repeated until a confirmation from each client computer system is received.

18. The method of claim **15** wherein the step of transmitting is periodically repeated until a time out has occurred.

19. A method in a server computer system for transmitting data to a plurality of client computer systems, the server computer system having a broadcast transmission mechanism and having a point-to-point transmission mechanism, the broadcast transmission mechanism for transmitting data from the server computer system to the client computer system but cannot transmit data from the client computer systems to the server computer system, each client computer system for receiving data transmitted through the point-to-point transmission mechanism, only some of the client computer systems capable of receiving data transmitted through the broadcast transmission mechanism, the method comprising the steps of:

transmitting the data using the broadcast transmission mechanism;

transmitting the data using the point-to-point transmission mechanism to each client computer system that is not capable of receiving the data transmitted through the broadcast transmission mechanism;

receiving a confirmation from each client computer system that received the data transmitted through the broadcast transmission mechanism; and

when confirmations have not been received from all the client computer system that are capable of receiving data transmitted through the broadcast transmission mechanism, re-transmitting the data using either the broadcast transmission mechanism or the point-to-point transmission mechanism so that each client computer system that is capable of receiving the data transmitted through the broadcast transmission mechanism receives the transmitted data.

20. A computer-readable medium containing instructions for causing a server computer system to transmit data to a plurality of client computer systems, the server computer system having a broadcast transmission mechanism and having a point-to-point transmission mechanism, the broadcast transmission mechanism for transmitting data from the server computer system to the client computer system but cannot transmit data from the client computer system to the server computer system, each client computer system for receiving data transmitted through the point-to-point transmission mechanism, only some of the client computer systems capable of receiving data transmitted through the broadcast transmission mechanism, by:

transmitting the data using the broadcast transmission mechanism;

transmitting the data using the point-to-point transmission mechanism to each client computer system that is not capable of receiving the data transmitted through the broadcast transmission mechanism;

receiving a confirmation from each client computer system that received the data transmitted through the broadcast transmission mechanism; and

when confirmations have not been received from all the client computer system that are capable of receiving data transmitted through the broadcast transmission mechanism, re-transmitting the data using either the broadcast transmission mechanism or the point-to-point transmission mechanism so that each client computer system that is capable of receiving the data transmitted through the broadcast transmission mechanism receives the transmitted data.

21. A computer system for confirming receipt of data broadcast by a server computer system, the broadcasted data being broadcasted using a broadcast transmission mechanism for transmitting data unidirectionally from the server computer system, comprising:

a receiving component that, when the computer system does not have a connection established with the server computer system, receives the broadcasted data and stores an indication that the broadcasted data has been received; and

a confirmation component that, when the computer system later establishes a connection with the server computer system, checks for presence of the stored indication that the broadcasted data has been received; and when the stored indication is present, sends a confirmation that the computer system received the broadcasted data to the server computer system using the established connection.

22. The computer system of claim **21** including a connection component that, after storing the indication, establishes a connection with the server computer system.

23. The computer system of claim **22** wherein the connection is established with a point-to-point transmission mechanism.

24. The computer system of claim **21** wherein when the stored indication is not present, the confirmation component sends an indication that the client computer system has not received any broadcasted data to the server computer system via the established connection.

25. The computer system of claim **21** wherein the server computer system resends the broadcasted data using the broadcast transmission mechanism.

26. The computer system of claim **21** wherein the server computer system resends the broadcasted data using the established connection.

* * * * *

US005666293A

United States Patent [19]

Metz et al.

[11] Patent Number: 5,666,293

[45] Date of Patent: Sep. 9, 1997

[54] **DOWNLOADING OPERATING SYSTEM SOFTWARE THROUGH A BROADCAST CHANNEL**

[75] Inventors: **Erik C. Metz**, Bowie; **Henry G. Hudson, Jr.**, Annapolis, both of Md.; **John W. Darr, Jr.**, Great Falls, Va.

[73] Assignee: **Bell Atlantic Network Services, Inc.**, Arlington, Va.

[21] Appl. No.: **498,265**

[22] Filed: **Jul. 3, 1995**

Related U.S. Application Data

[63] Continuation-in-part of Ser. No. 380,755, Jan. 31, 1995, and Ser. No. 250,791, May 27, 1994.

[51] **Int. Cl.**[6] **H04H 1/00**

[52] **U.S. Cl.** **395/200.5**; 455/3.1; 455/4.1; 455/4.2; 455/5.1; 348/7; 348/10; 348/12

[58] **Field of Search** ..... 364/514 C; 380/10, 380/20; 370/112, 118; 348/7, 10, 12; 455/3.1, 4.1, 4.2, 5.1, 6.1

[56] **References Cited**

U.S. PATENT DOCUMENTS

Re. 34,611	5/1994	Fenwick et al. .	
4,506,387	3/1985	Walter .	
4,527,194	7/1985	Sirazi .	
4,623,905	11/1986	Ichihashi et al. .	
4,623,920	11/1986	Dufresne et al. .	
4,677,685	6/1987	Kurisu .	
4,700,386	10/1987	Kohn .	
4,706,121	11/1987	Young .	
4,709,418	11/1987	Fox et al. .	
4,712,239	12/1987	Frezza et al. .	
4,816,905	3/1989	Tweedy et al. .	
4,829,372	5/1989	McCalley et al. .	
4,894,714	1/1990	Christis .	
4,912,552	3/1990	Allison, III et al. .	
4,920,432	4/1990	Eggers et al. .	
4,947,244	8/1990	Fenwick et al. .	
4,949,187	8/1990	Cohen .	
4,963,995	10/1990	Lang .	
4,982,430	1/1991	Frezza et al. .	
5,003,591	3/1991	Kauffman et al.	380/10
5,010,499	4/1991	Yee .	
5,027,400	6/1991	Baji et al. .	

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

3149992	6/1991	Japan .
01-288421	9/1991	Japan .
94/23537	10/1994	WIPO .

OTHER PUBLICATIONS

Hambley, Allan R., "Communication Systems", pp. 8–10 1990.

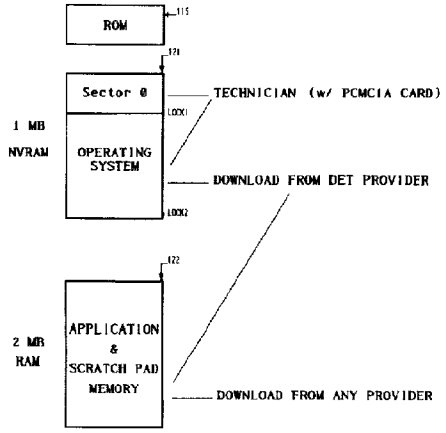
Gelman et al., A Store-and-Forward Architecture for Video-on-Demand Service, International Conference on Communications, Denver, Jun. 23, 1991; Communications: Rising to the Heights; vol. 2 of 3; pp. 842–846.

Primary Examiner—Ellis B. Ramirez
Assistant Examiner—Patrick Assouad
Attorney, Agent, or Firm—Lowe, Price, LeBlanc & Becker

[57] **ABSTRACT**

Set-top terminals utilized in broadband broadcast networks are becoming increasingly intelligent. Upgrading the operation of such terminals periodically requires upgrading the software, particularly the operating system, of the programmable processor which controls the terminal operation. To facilitate frequent upgrades, the network will carry a cyclic broadcast of a packetized data file containing the operating system. Periodically, a terminal will capture and store the broadcast operating system. In the preferred embodiment, the broadcast includes operating system files for a number of different terminal types and data identifying the current broadcast version of the operating system for each type of terminal. The terminal will check the broadcast version number for its terminal type operating system. If the broadcast version number differs from the version number for the operating system the terminal currently is running, then the terminal will capture only the file containing the operating system for the corresponding terminal type.

45 Claims, 9 Drawing Sheets



U.S. PATENT DOCUMENTS

5,051,822	9/1991	Rhoades .	
5,057,932	10/1991	Lang .	
5,058,160	10/1991	Banker et al. .	
5,104,125	4/1992	Pocock et al. .	
5,105,268	4/1992	Yamanouchi et al. .	
5,119,188	6/1992	McCalley et al. .	
5,121,476	6/1992	Yee .	
5,130,792	7/1992	Tindell et al. .	
5,132,992	7/1992	Yurt et al. .	
5,133,079	7/1992	Ballantyne et al. .	
5,136,411	8/1992	Paik et al. .	
5,140,417	8/1992	Tanaka et al. .	
5,142,680	8/1992	Ottman et al. .	
5,166,886	11/1992	Molnar et al. .	
5,168,353	12/1992	Walker et al. .	
5,172,413	12/1992	Bradley et al. .	
5,181,107	1/1993	Rhoades .	
5,189,673	2/1993	Burton et al. .	
5,192,999	3/1993	Graczyk et al. .	
5,223,924	6/1993	Strubbe et al. .	
5,231,494	7/1993	Wachob .	
5,239,540	8/1993	Rovira et al. .	
5,247,347	9/1993	Litteral et al. .	
5,247,364	9/1993	Banker et al. .	
5,249,044	9/1993	Von Kohorn .	
5,253,275	10/1993	Yurt et al. .	
5,282,028	1/1994	Johnson et al. .	
5,315,392	5/1994	Ishikawa et al. .	
5,317,391	5/1994	Banker et al. .	
5,335,277	8/1994	Harvey et al. .	
5,341,425	8/1994	Wasilewski	380/20
5,341,474	8/1994	Gelman et al. .	
5,373,288	12/1994	Blahut .	
5,379,421	1/1995	Palazzi, III et al. .	
5,400,401	3/1995	Wasilewski	380/9
5,410,326	4/1995	Goldstein .	
5,418,782	5/1995	Wasilewski	370/73
5,421,017	5/1995	Scholz et al. .	
5,440,632	8/1995	Bacon et al.	380/20
5,441,389	8/1995	Blahut et al. .	
5,448,568	9/1995	Delpuch et al.	372/94.2
5,548,532	8/1996	Menand et al.	364/514 C
5,553,311	9/1996	McLaughlin et al.	395/884
5,563,648	10/1996	Menand et al.	348/13

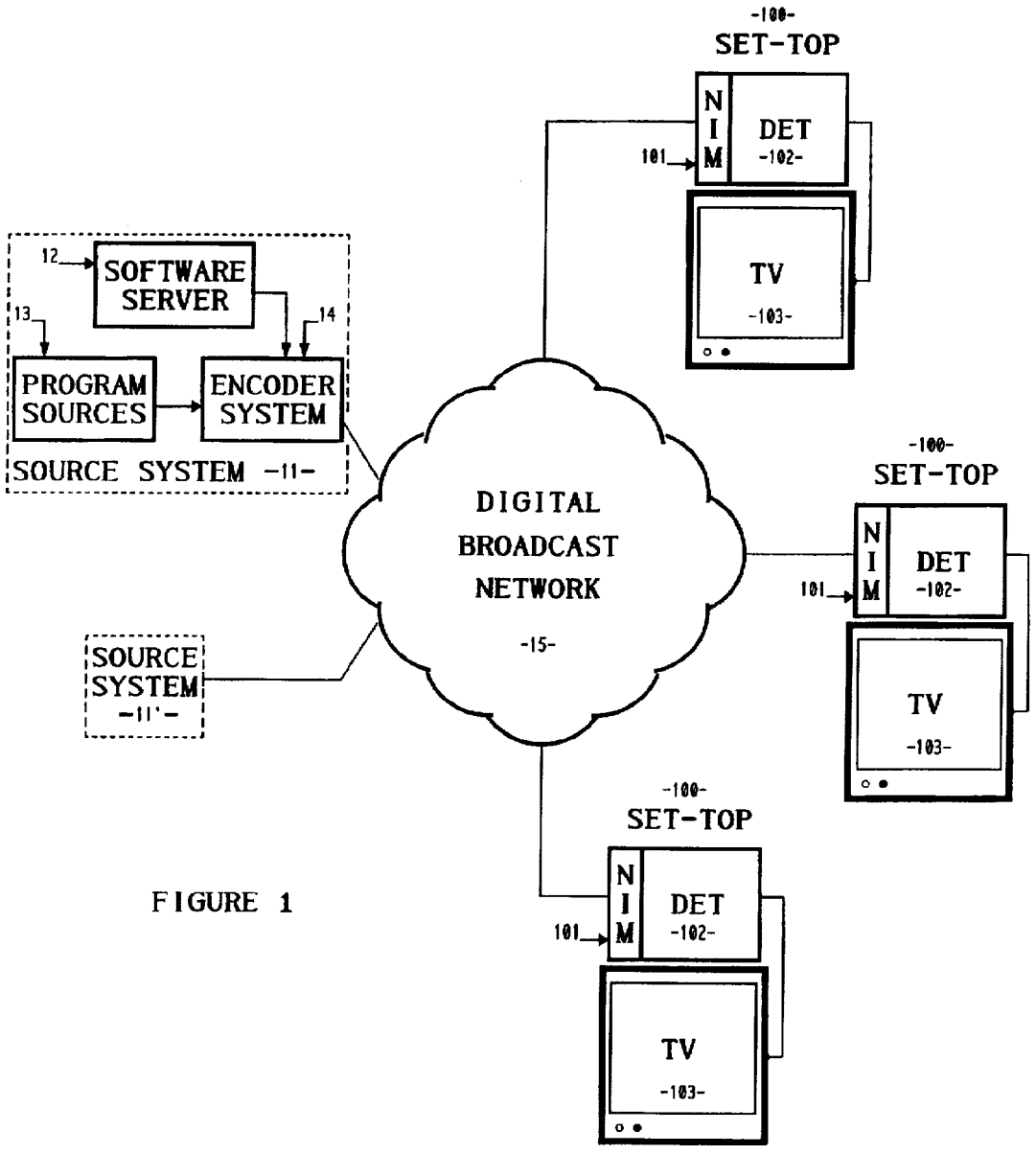

-100-
SET-TOP
N I M
DET -102-
101
TV -103-
12
SOFTWARE SERVER
13
14
PROGRAM SOURCES
ENCODER SYSTEM
SOURCE SYSTEM -11-
DIGITAL BROADCAST NETWORK -15-
SOURCE SYSTEM -11'-
-100-
SET-TOP
N I M
DET -102-
101
TV -103-
-100-
SET-TOP
N I M
DET -102-
101
TV -103-

FIGURE 1

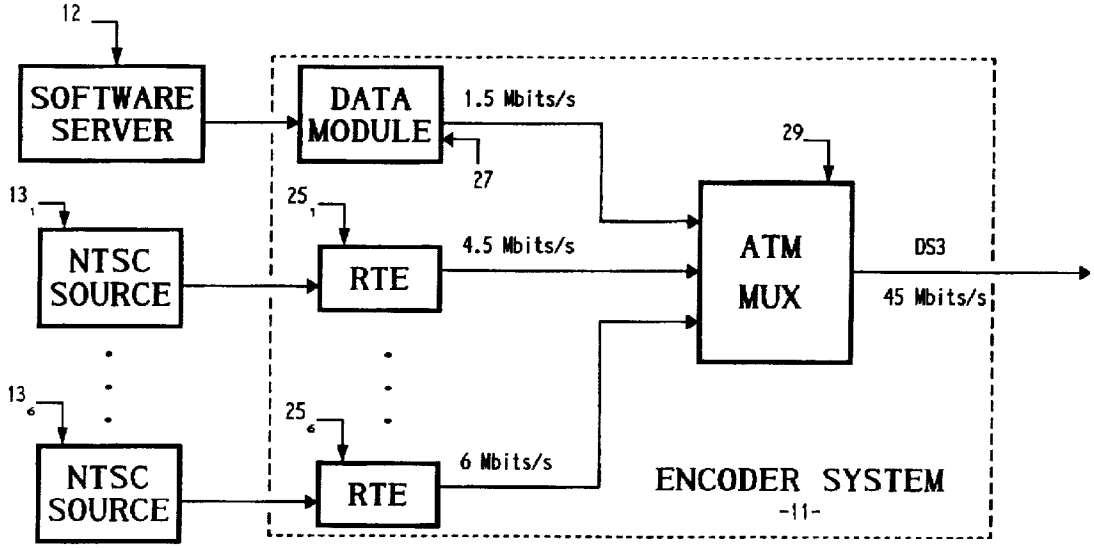

12
SOFTWARE SERVER
DATA MODULE
1.5 Mbits/s
27
29
13 1
NTSC SOURCE
25 1
RTE
4.5 Mbits/s
ATM MUX
DS3
45 Mbits/s
13 6
NTSC SOURCE
25 6
RTE
6 Mbits/s
ENCODER SYSTEM
-11-

FIGURE 2

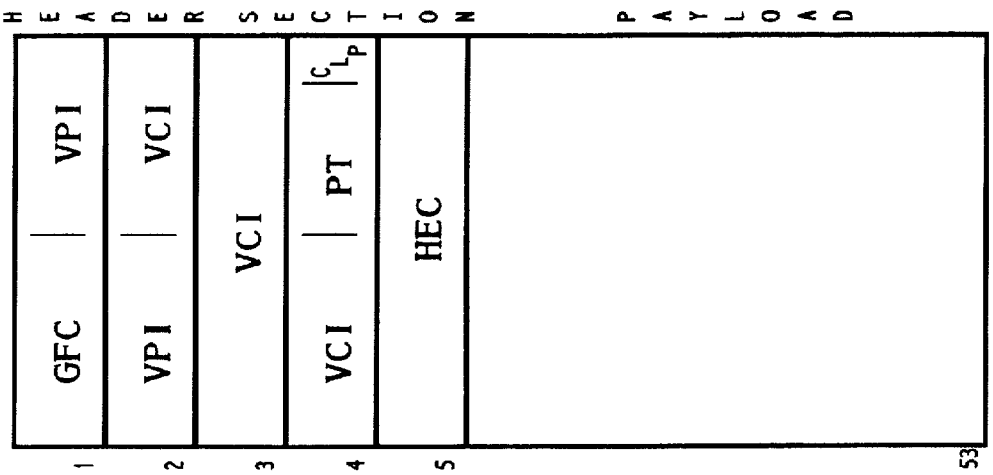


FIGURE 4

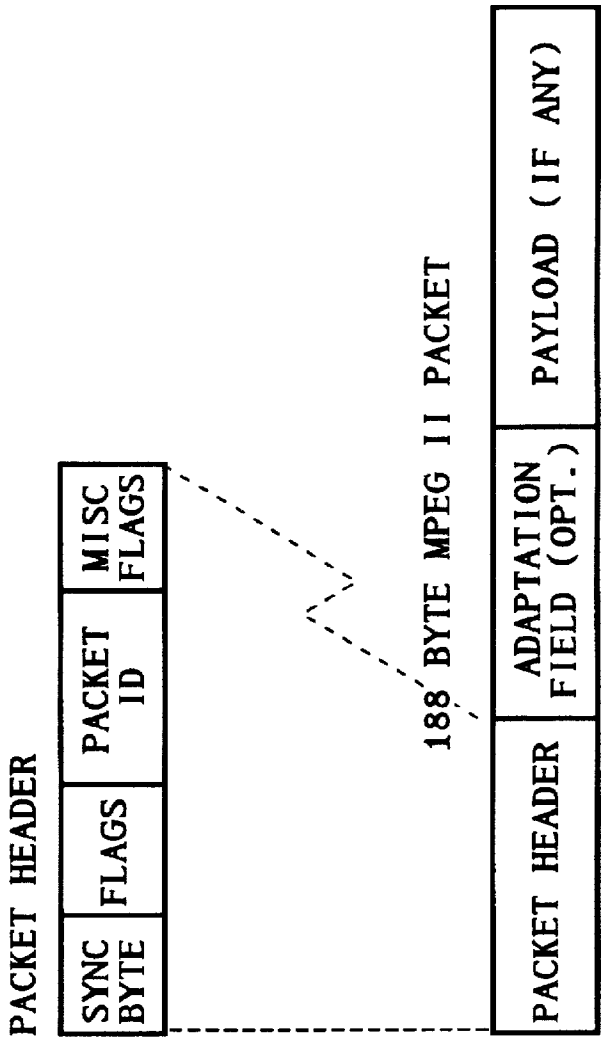


FIGURE 3

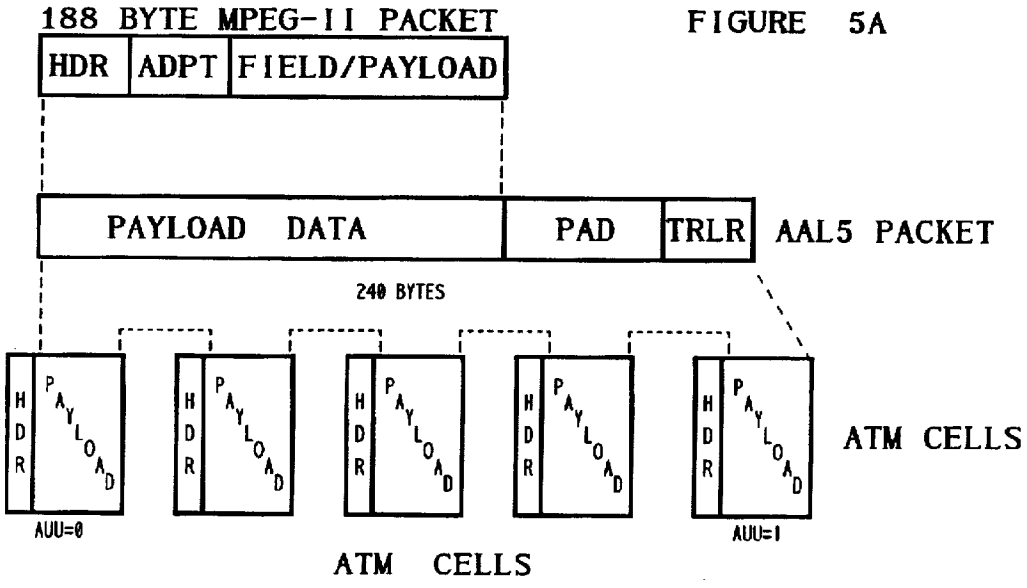
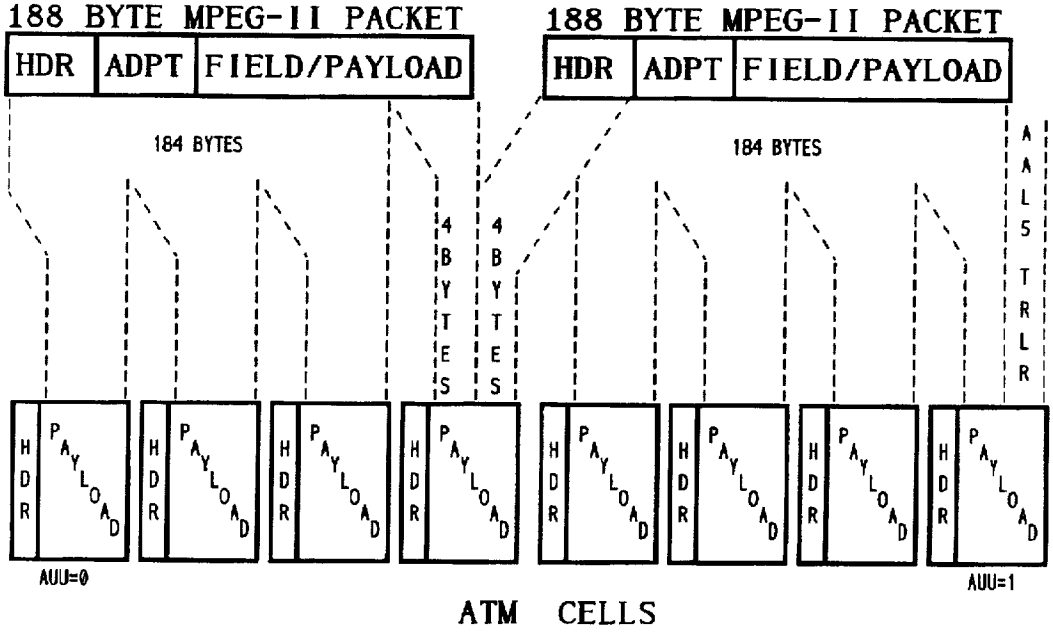


FIGURE 5A

FIGURE 5B



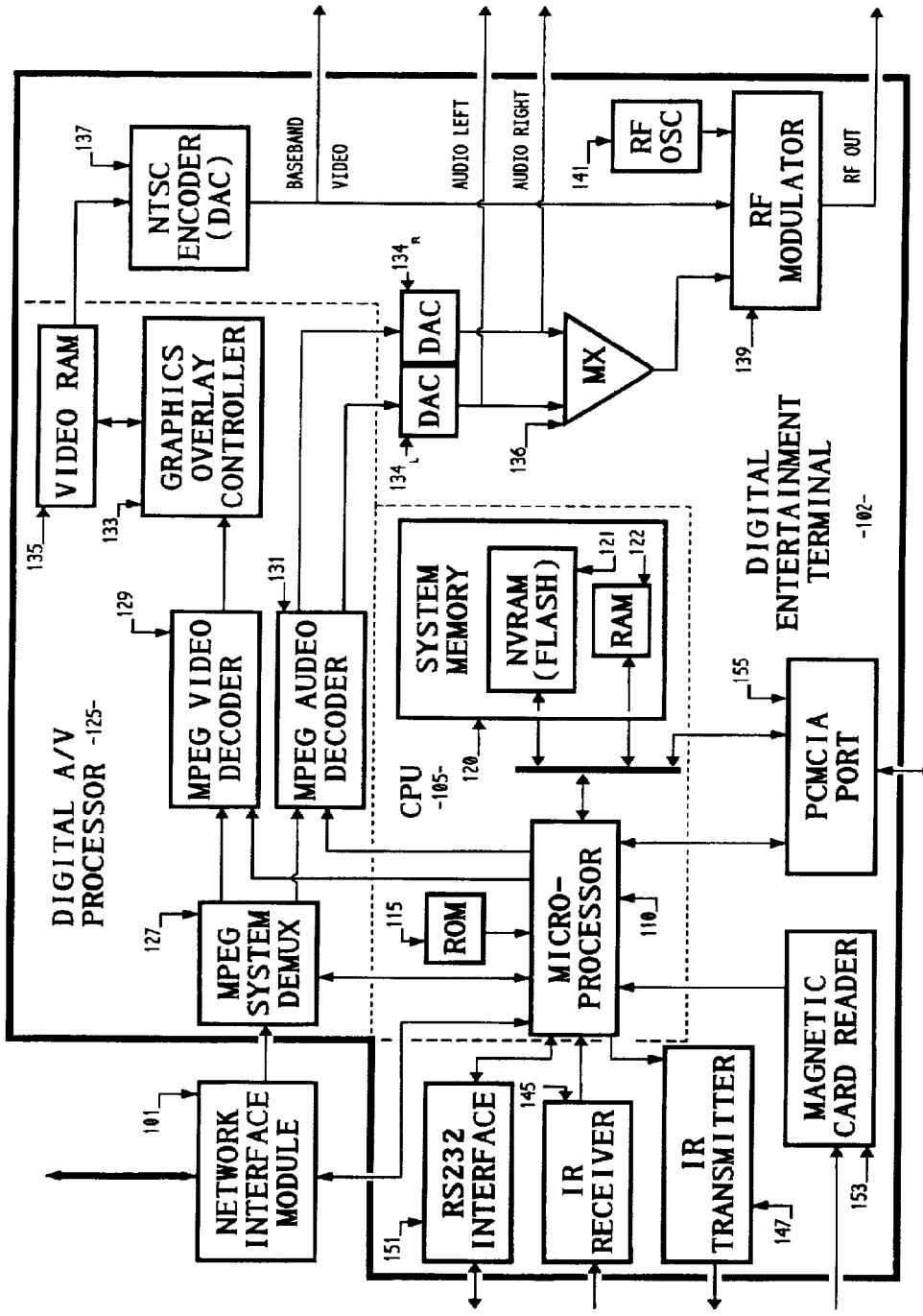

DIGITAL A/V PROCESSOR -125-
NETWORK INTERFACE MODULE
101
MPEG SYSTEM DEMUX
127
MPEG VIDEO DECODER
129
MPEG AUDEO DECODER
131
VIDEO RAM
135
GRAPHICS OVERLAY CONTROLLER
133
NTSC ENCODER (DAC)
137
BASEBAND VIDEO
AUDIO LEFT
AUDIO RIGHT
DAC
DAC
134n
134L
MX
136
RF OSC
141
RF MODULATOR
139
RF OUT
CPU -105-
ROM
115
MICRO-PROCESSOR
110
SYSTEM MEMORY
120
NVRAM (FLASH)
121
RAM
122
DIGITAL ENTERTAINMENT TERMINAL -102-
PCMCIA PORT
155
RS232 INTERFACE
151
IR RECEIVER
145
IR TRANSMITTER
147
MAGNETIC CARD READER
153
SET-TOP -100-

FIGURE 6

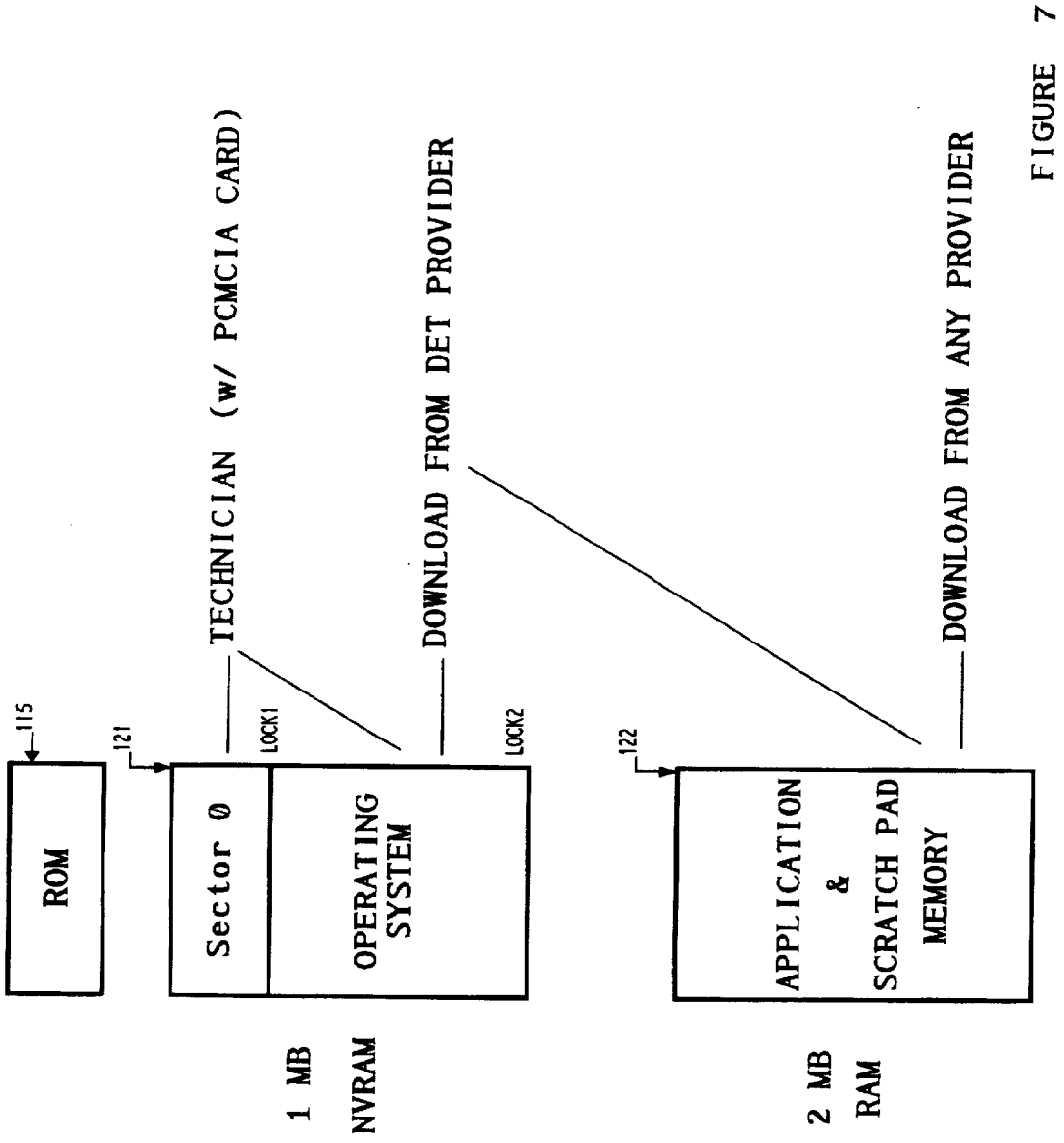

115
ROM
121
Sector 0
LOCK1
OPERATING SYSTEM
LOCK2
1 MB NVRAM
TECHNICIAN (w/ PCMCIA CARD)
DOWNLOAD FROM DET PROVIDER
122
APPLICATION & SCRATCH PAD MEMORY
2 MB RAM
DOWNLOAD FROM ANY PROVIDER

FIGURE 7

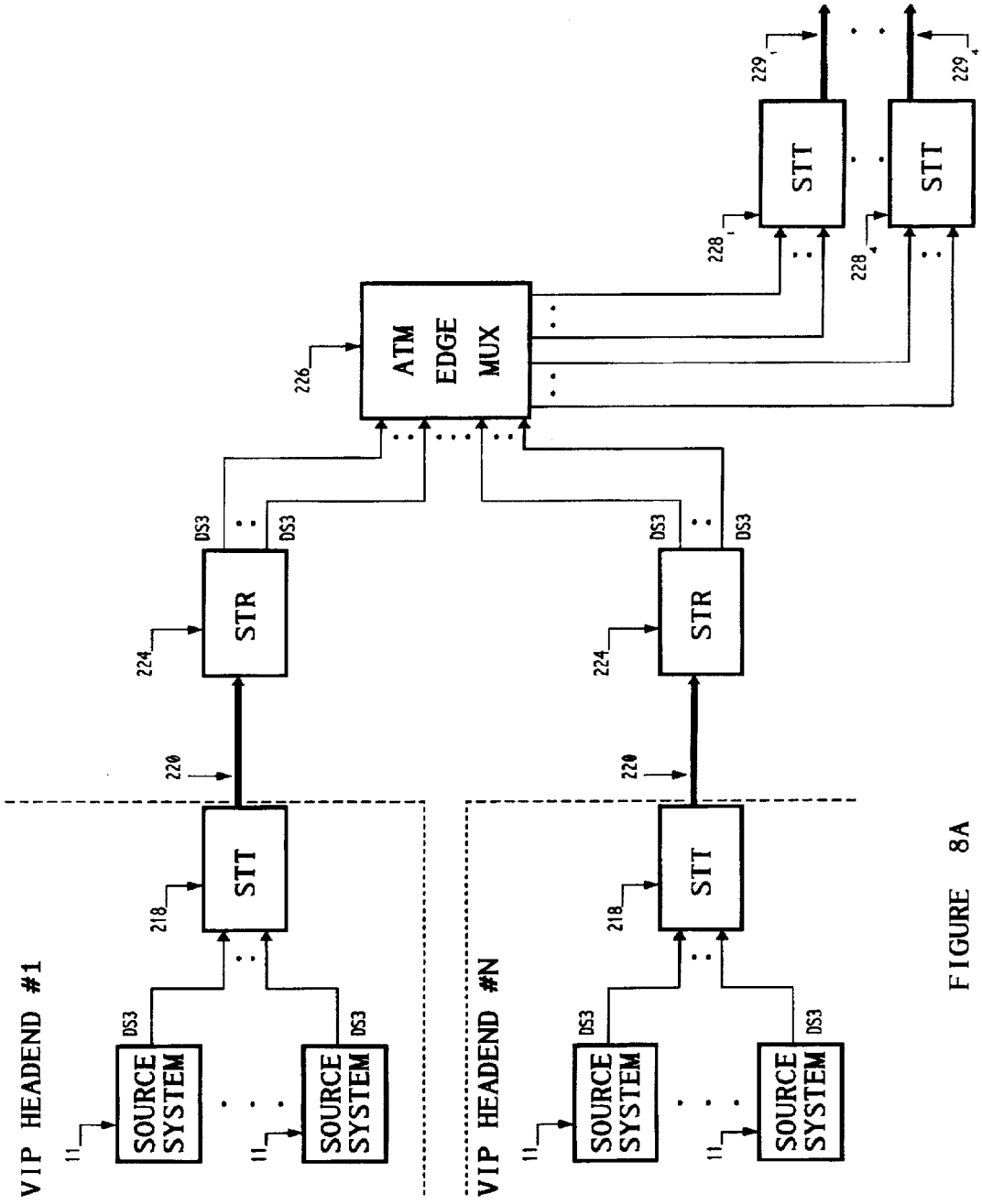

VIP HEADEND #1
VIP HEADEND #N
SOURCE SYSTEM
DS3
STT
218
220
STR
224
ATM EDGE MUX
226
228
229

FIGURE 8A

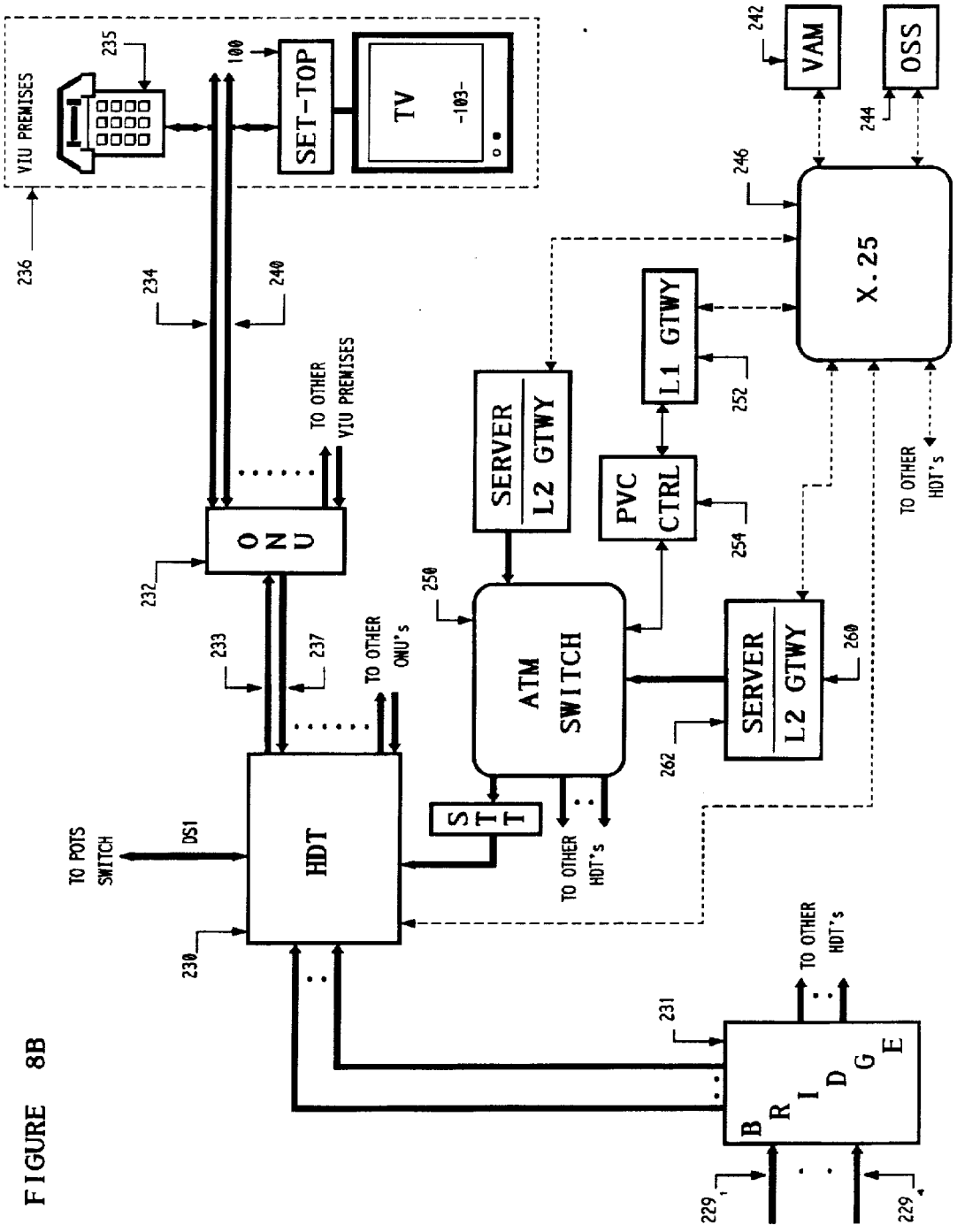

VIU PREMISES
235
100
SET-TOP
TV
-103-
236
234
240
TO OTHER VIU PREMISES
ONU
232
233
237
TO OTHER ONU's
HDT
TO POTS SWITCH
DS1
230
STT
TO OTHER HDT's
ATM SWITCH
250
SERVER L2 GTWY
PVC CTRL
254
L1 GTWY
252
X.25
246
VAM
242
OSS
244
TO OTHER HDT's
SERVER L2 GTWY
260
262
TO OTHER HDT's
231
BRIDGE
229
229

FIGURE 8B

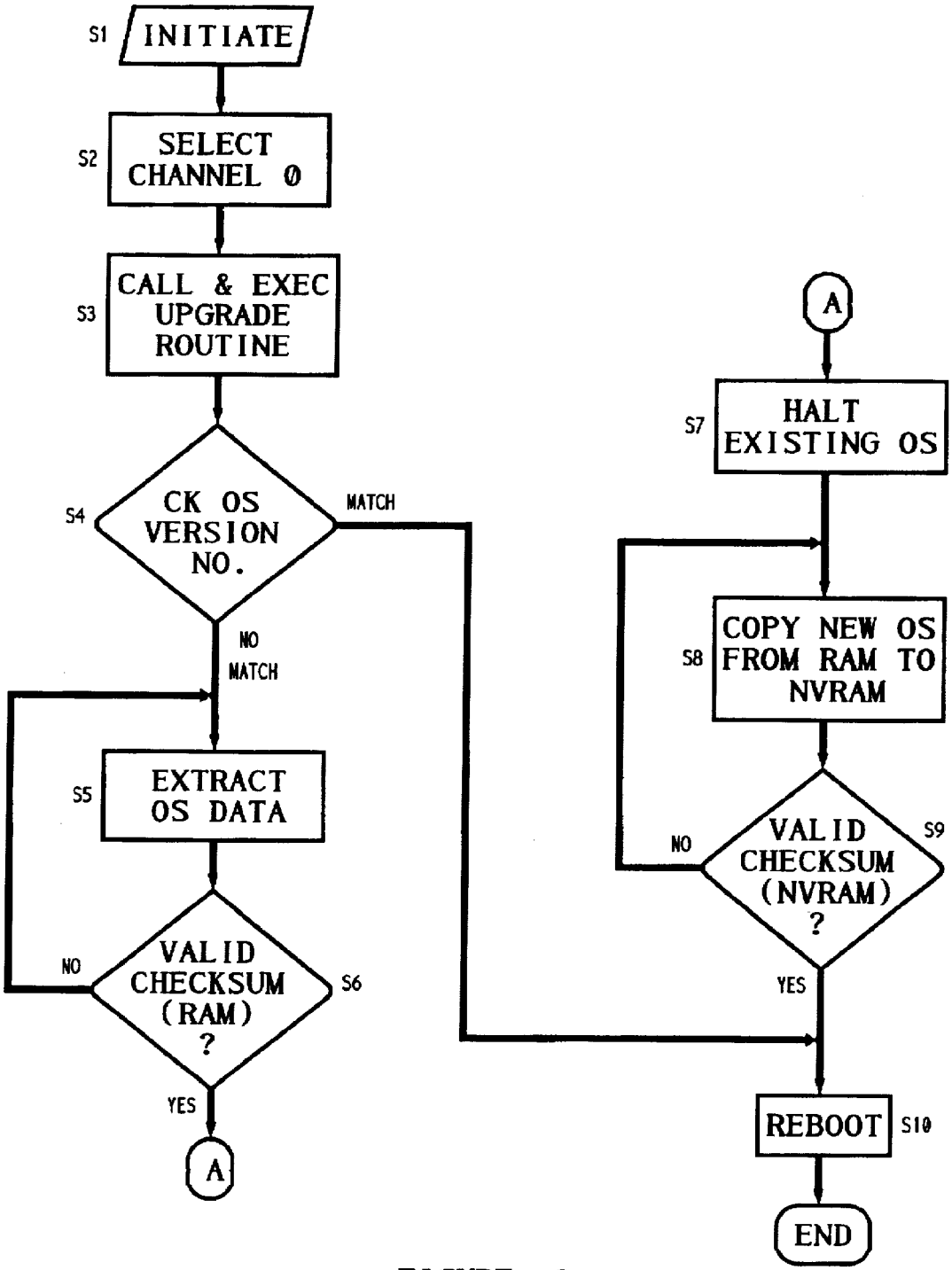

S1
INITIATE
S2
SELECT CHANNEL 0
S3
CALL & EXEC UPGRADE ROUTINE
S4
CK OS VERSION NO.
MATCH
NO MATCH
S5
EXTRACT OS DATA
S6
VALID CHECKSUM (RAM) ?
NO
YES
A
A
S7
HALT EXISTING OS
S8
COPY NEW OS FROM RAM TO NVRAM
S9
VALID CHECKSUM (NVRAM) ?
NO
YES
REBOOT
S10
END

FIGURE 9

DOWNLOADING OPERATING SYSTEM SOFTWARE THROUGH A BROADCAST CHANNEL

CROSS REFERENCE TO RELATED APPLICATION

This application is a Continuation-In-Part of U.S. patent application Ser. No. 08/380,755 filed Jan. 31, 1995 which is a Continuation-In-Part of U.S. patent application Ser. No. 08/250,791 filed May 27, 1994, the disclosures of both of which are incorporated herein entirely by reference.

TECHNICAL FIELD

The present invention relates to a programmable set-top terminal, typically comprising a network interface module (NIM) and a digital entertainment terminal (DET), for use in digital video program distribution networks and to systems and methods for dynamically downloading operations system software to such a terminal.

Background Art

Set top terminal devices commonly in use in cable television systems today have a number of limitations. First, the devices are limited to processing of analog television signals. Also, cable television terminal devices are generally "dumb" devices having a limited set of functionalities constrained by the hard wired programming of the internal micro-processor controlled device. Essentially all cable television terminal devices respond to a selection input from the subscriber, tune to a selected channel available on the cable television network, decode the video program material if scrambled, and provide output signals compatible with a standard television receiver.

Enhanced cable television terminals provide some additional features, such as graphics overlay capability and two way communication of control signalling to and from headend terminal devices. Although such improved terminals facilitate some enhanced services, such as home shopping and purchasing, the performance of these cable television set-top terminals is still limited to analog decoding. Also the range of services is still limited by the hard wired capabilities of the microprocessor within the set-top terminal devices.

Proposals have been made to download computer executable code over cable television networks. In particular, U.S. Pat. Nos. 5,051,822 and 5,181,107 both to Rhoades disclose a terminal device connectable to a cable television network and a telephone line. A subscriber requests a video game or other software stored in a remotely located software storage center by operating the terminal to establish a bi-directional telephone link with the remote storage center. The center transmits the encoded software program together with the terminal identification code as a digital bit stream over a television broadcast channel. The terminal requesting the software monitors all digital bit streams on the broadcast channel but receives only the software program addressed to it, i.e. only after identification code validation occurs. Once reception of all the software data is complete, the terminal acknowledges receipt to the remote storage center and drops the telephone line. The encoded software program is decoded, and the terminal provides a display informing the subscriber that the game or other program is ready for use. The terminal also offers the subscriber the means to interact with the software, e.g. play the game, using contemporary gaming control or input devices. While the Rhoades terminal

structure does provide enhanced capabilities, such as video games and home shopping, the display functionality controlled by the downloaded software is limited to computer displays generated in response to the software, there is no direct interaction of the received software with any video program carried on the cable network. The downloaded software does not control further interactions with the storage center. Also, the video transmissions on the cable system are analog, and a separate telephone connection is required for selection inputs to the central storage facility. Furthermore, the terminal device apparently can receive software from the storage center of only one service provider.

Some prior art systems do permit downloading into the cable television decoder itself, however, it is believed that this downloading of information into the decoder has been limited to information controlling the decoding of the television program signals, e.g. a key word used in a descrambling algorithm. Dufresne et al., in U.S. Pat. No. 4,623,920 teach a specific scheme for addressing data transmissions over a cable television network to groups of terminals or to individual terminals. The addressed data sent from the head end can include an option table of signals for controlling descrambling of available television programs, data to enable operation of a cable TV converter, or software for operating a peripheral microcomputer separate from the cable television terminal device. The Dufresne et al. terminal is limited to reception of data from only one service provider, i.e. the provider operating the cable TV network. Also, the services provided through the terminal are limited in that the downloaded data apparently does not alter or control the terminal functionality for further interactions with the provider through the network.

Recently, several different wideband digital distribution networks have been proposed for offering subscribers an array of video services, such as Video On Demand. The following U.S. Pat. Nos. disclose representative examples of such digital video distributions networks: 5,253,275 to Yurt et al., 5,132,992 to Yurt et al., 5,133,079 to Ballantyne et al., 5,130,792 to Tindell et al., 5,057,932 to Lang, 4,963,995 to Lang, 4,949,187 to Cohen, 5,027,400 to Baji et al., and 4,506,387 to Walter. The terminal devices in these digital networks are still limited functionality devices. In these networks, the digital terminal devices still only receive selection inputs, transmit selection signals upstream to the source of the video materials, receive downstream video transmissions, decompress the digitized video materials and convert to analog form, and provide appropriate signals to a television receiver. One example of such a digital video distribution network and the terminal device for such a network, disclosed in Litteral et al. Pat. No. 5,247,347, will be described in more detail below.

U.S. Pat. No. 5,247,347 to Litteral et al. discloses an enhanced public switched telephone network which also provides a video on demand service to subscribers over the public switched telephone network. A menu of video programming information is displayed at the subscriber's premises by a set-top terminal and a TV set. The subscriber may transmit ordering information via the public switched telephone network to the independent video information providers. Video programming may be accessed and transmitted to the subscriber directly from a video information provider (VIP) or through a video buffer located at a central office (CO) serving the subscriber.

Connectivity between the central office and the subscriber for transmission of video data is provided by an asymmetrical digital subscriber line (ADSL) system. ADSL interface

units at the central office multiplex digital video information with voice information to be transmitted to the subscriber and support two-way transmission between the subscriber's line and the X.25 packet data network of one or more control channels. A complimentary ADSL interface unit at the subscriber's premises separates downstream video control signals and voice telephone signals from the line and multiplexes upstream control signals and voice telephone signals onto the line. The ADSL interface on the subscriber premises supplies the broadband digital data stream recovered from the transmission over the subscriber loop to a decoder unit in the set-top terminal. The decoder unit decompresses the audio and video data, and converts the digital audio and video to corresponding analog signals. The decoder can supply baseband analog audio and video signals to a television receiver, or these analog signals can be modulated to a standard television channel frequency for use by the television receiver.

The above detailed discussion of the Litteral et al. system shows that prior art digital distribution networks offer enhanced video services, but the terminal device functionality is still limited to program selection, decoding and display.

A number of suggestions have been made in the press regarding arrays of different services which will become available through broadband digital networks now popularly referred to as the "Information Super Highway". If a different VIP were to offer a different service, the VIP can limit the service to an interactivity with the subscriber essentially corresponding to the functionality available in the terminal device. This approach, however, limits the functional capabilities the new VIP may choose for the different service. Alternatively, the subscriber must buy another terminal device programmed or wired to function in accord with the VIP's new service. This second approach, however, forces the subscriber to purchase and connect up a different terminal device for each different service subscribed to.

From the above discussion it becomes clear that a need exists in the art for set-top terminal devices which process compressed, broadband digital audio/video information and are readily adaptable to perform a variety of related functionalities, as needed to facilitate a range of audio/video and interactive services offered by a large number of information providers.

In the 08/250,791 grandparent application cited above, it was suggested that software could be downloaded into the digital set-top terminal through a point-to-point connection through a digital broadband network, e.g. similar to that of Litteral et al. As disclosed therein, the software included at least customized applications programs for controlling terminal operation in a manner specified by an individual information provider. It was also suggested that at least one party would operate a server to download operations system upgrades through a point-to-point connection. Point-to-point connections through broadband digital networks are relatively expensive, and some digital networks under development will have broadcast channels, but at least initially, will not offer point-to-point connections. The disclosure in the 08/250,791 application did not address problems of downloading software to terminals through digital broadcast networks.

In the 08/380,755 parent application cited above, it was suggested that software, specifically software related to channel mapping functionalities and navigation through broadcast services, could be downloaded into the digital set-top terminal through a data carousel type cyclical broadcast. Such downloaded software consisted of one or more applications intended for wide general availability. The digital type set-top devices receiving such software were intended as open interface devices to which any provider offering such a download service could download the relevant data and executable code.

An operating system includes programming to control internal operations of the control processor, such as those necessary to execute specific types of communications over the network, graphics drivers, etc. The operating system typically allows the set-top to run a variety of downloaded applications programs, preferably made available by a number of service providers. It is desirable to periodically update the operating system software, as improvements are developed, without having a technician manually service each terminal. The downloading of an operating system program for running the terminal device raises a more complicated set of problems relating to who can download such software to which types of terminals.

Access to the ability to modify the operating system must be carefully controlled. If access were open, an unscrupulous party could write a destructive operating system, e.g. that would allow the terminal to access only one provider's services or that might cause the terminal to begin upstream transmissions in some manner which would disrupt upstream transmissions of other terminals. The downloaded operating system would need to correspond to the particular type of set-top terminal to insure compatibility. Also, the downloading of the operating system must be particularly error free to insure that errant reception and overwriting of operation system software does not in some corrupt or disable terminal operation.

A need therefore still exists to reliably and securely download operating system software to the digital set-top terminal through a widely accessible broadcast channel.

DISCLOSURE OF THE INVENTION

The present invention addresses the above noted needs by providing methods, systems and terminal device structures for downloading operating system software to programmable set-top terminal devices through digital broadcast channels.

In one aspect, the invention contemplates a set-top terminal device to which new operating system software can be downloaded through one of the broadcast channels. The terminal device includes a network interface module. This module couples the terminal to a communication network. From the network, the interface module receives at least selected ones of a plurality of broadcast digital broadband channels. One or more of the broadcast channels carries audio/video program information in compressed, digital form in packets of a standardized format. Also, one of the broadcast channels carries cyclically repetitive transmissions of the operating system software in packets of the standardized format.

The set-top terminal also includes a digital entertainment terminal. The digital entertainment terminal includes an audio/video processor for processing the compressed, digital audio/video program information from a selected broadcast channel. The digital entertainment terminal also includes a memory and a remote control or the like for supplying inputs from a user to the digital entertainment terminal. A control processor captures the operating system software from one of the selected digital broadband channels. The control processor loads the captured operating system software into the memory and begins operation in accord with the oper-

ating system software. For example, using the operating system software in the memory, the control processor controls the network interface module and the audio/video processor in response to the user inputs.

Another aspect of the invention relates to a communication system including a network for broadcasting the channels to a plurality of terminals, similar to the set-top terminal discussed above. This system includes a source system supplying program material and software for broadcast through a digital network. The source system comprises a program source supplying a broadband program signal and a software server cyclically outputting a data file containing an operating system. An encoder system packetizes the broadband program signal and the data file in digital packets of the standard format.

In the preferred implementation, the encoder processes analog audio/video signals to digitize, compress and packetize the program information in accord with the moving pictures expert group (MPEG) standard. The preferred network utilizes Asynchronous Transfer Mode (ATM) transport. The encoder system therefore includes an ATM multiplexer for adapting the MPEG packets into ATM cells and combining ATM cells from one or more programs together with ATM cells containing the operating system into a stream for transport through the ATM broadcast network.

As digital networks develop and remain in wide use over a period of time, the set-top terminal device will essentially become a piece of consumer electronics equipment. At such a time, different end users will obtain set-top terminals of different types from a number of different providers and will connect different types of set-top terminals to the same digital network. Different types of set-top terminals will utilize different operating systems. A further aspect of the invention therefore relates to broadcasting a plurality of different operating systems for correspondingly different types of set-top terminals. Each type of set-top terminal will identify the correct operating system from among the plurality broadcast and capture only that operating system.

Another feature of the invention relates to identification of the need for a particular set-top terminal to upgrade its operating system. Specifically, each operating system for a particular type of set-top terminal has a version number. The set-top terminal stores a version number for the operating system that it currently is running, and the broadcast data stream will include data identifying the version number of the operating system being broadcast for the particular type of terminal. The set-top terminal actually captures an operating system from the broadcast if the broadcast version number is different (e.g. higher or lower) than the number of the version that terminal is currently running.

In accord with the present invention, the operating system upgrade process can begin automatically, or a user can manually trigger the upgrade process. For automatic activation, a processor in the set-top terminal will monitor some periodic occurrence, such as the passage of some time interval or cycles of turn-off by a user. For manual activation, the user may call up a menu display by the set-top terminal and select the operating system upgrade from the menu.

Applications software can be downloaded to the set-top via the network. The set-top may capture a desired application from a digital broadcast channel in a manner similar to that used to acquire the new operating system. Alternatively, the user may establish a point-to-point broadband call to an interactive service provider's system, in which case, the service provider's system downloads an application to control further interactivity via the point-to-point link.

The present invention may be utilized on a variety of different types of broadcast networks, particularly those carrying digitized and compressed broadcast programming. Several networks are cited, and a preferred network is disclosed in detail. The preferred digital network includes a system of optical fibers for broadcasting the digital packets from the encoder system to a plurality of host digital terminals. Each host digital terminal routes selected digital broadcast channels to a group set-top terminals connected thereto.

Additional objects, advantages and novel features of the invention will be set forth in part in the description which follows, and in part will become apparent to those skilled in the art upon examination of the following or may be learned by practice of the invention. The objects and advantages of the invention may be realized and attained by means of the instrumentalities and combinations particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF DRAWINGS

FIG. **1** illustrates a digital broadcast system utilizing the operating system download of the present invention.

FIG. **2** shows a software server, program sources and an encoder system used in the network of FIG. **1**.

FIG. **3** illustrates an exemplary structure of an MPEG II type data packet.

FIG. **4** shows an exemplary structure of an ATM cell.

FIG. **5A** illustrates a five-cell adaptation for mapping an MPEG II packet into ATM cells.

FIG. **5B** illustrates an eight-cell adaptation for mapping two MPEG II packets into ATM cells.

FIG. **6** illustrates a digital set-top terminal device in accord with the present invention.

FIG. **7** shows a memory layout for the digital entertainment terminal and an associated diagram of functions involved in memory management and software downloading in accord with the present invention.

FIGS. **8A** and **8B** together depict a block diagram of a full service digital broadband network in accord with a preferred embodiment of the present invention.

FIG. **9** is a flow chart illustrating an exemplary procedure for upgrading the operating system of the set-top terminal device using software downloaded through a broadcast channel.

BEST MODE FOR CARRYING OUT THE INVENTION

In the digital broadband networks of type under consideration here, each user has a set-top terminal device **100** (FIG. 1). The set-top device **100** includes a digital entertainment terminal (DET) **102** and a network interface module (NIM) **101**.

With the present invention the set-top terminal **100** (preferably the DET portion **102** thereof) receives and stores downloaded operating system software and application software. The terminal **100** can establish a point to point link to interactive equipment operated by a video information provider (VIP) and receive interactive applications software through the point to point link, as disclosed in the above incorporated 08/250,791 application. The key features of the present invention, however, relate to downloading an operating system through a broadcast channel, therefore the following description concentrates on a broadcast type network implementation and downloading of the operating system through a broadcast channel.

FIG. 1 is a high level functional diagram of a network providing digital broadcast services, preferably using ATM cell transport. The preferred embodiment illustrated in FIGS. 8A and 8B and discussed later utilizes end-to-end ATM transport, i.e. with ATM cells for at least the downstream broadband transmissions going all the way to the set-top terminal devices **100**. Other networks which may carry the operating system download in accord with the present invention, such as the hybrid-fiber-coax network shown in FIG. 4 of U.S. patent application Ser. No. 08/304,174, utilize ATM transport in a backbone portion of the network only and use some other transport technology for local loop distribution to the subscriber's terminal device. The software downloading techniques of the present invention can be applied to other such digital broadcast networks.

FIG. 1 therefore provides a generic illustration of the broadcast network **15**. As shown, the network **15** receives digitized data streams, preferably in ATM cell format, from one or more sources **11** operated by one or more information providers. In the later discussed preferred embodiment, local loop distribution utilizes switching nodes referred to as host digital terminals (HDT's) which transport ATM cell streams through to the relevant subscribers' set-top terminals **100**. In some forms of the network **15**, local loop distribution nodes may strip off the ATM cell headers and convert the payload data to some other format for actual transmission to the subscriber terminals. In the preferred embodiment, the local loop distribution network supplies the ATM cells from each broadcast to each set-top terminal **100** from which a subscriber requested the particular broadcast service.

Material intended for broadcast through the network is encoded and packetized in accord with a specified protocol or standard, such as DIGICIPHER™. The preferred embodiments utilize MPEG (moving pictures expert group) encoding. The source system **11** includes one or more program sources **14** and an encoder system **13** for encoding the program material in the desired standard format. Where the network utilizes another transport protocol such as ATM, the encoder also adapts the encoded information to the format utilized on the network **15**. In accord with the present invention, the source system **11** also includes a software server which supplies data to the encoder **14**.

As shown, a number of source systems **11**, **11'** supply digitized material to the digital network **15** for broadcast. One service provider may operate a number of the source systems to provide a desired number of broadcast programs or channels. Also, the network may offer a 'video dial tone' type service whereby a plurality of video information providers (VIPs) separately supply their own programming from one or more such sources. In the simplified example shown in FIG. **1**, source system **11** offers a plurality of broadcast programs from sources **13** and broadcasts software for the downloading service. Other source systems such as system **11'** may be identical to system **11** and offer both broadcast programming and software downloading, but most of the other systems **11'** will offer only broadcast programming. Source systems offering broadcast programs only will be similar in structure and operation to the system **11** discussed below in more detail with regard to FIG. **2**, but those systems **11'** will not include the software server and the associated element(s) of the encoder for processing the software.

In normal operation, the broadcast network supplies at least a selected program channel to the set-top terminal **100**. The set-top terminal processes information from the selected channel to produce signals capable of presenting information from that channel to a user in humanly perceptible form, e.g. to drive a standard television set **103** to display selected video programming. The NIM **101** provides the actual physical connection to the network and the transport protocol processing (e.g. ATM). The DET **102** performs the actual decoding to produce the output signals from the information. The DET **102** also includes the primary intelligent control processor for overall control of the operation of the set-top terminal **100**.

The DET portion of the set-top device **100** includes a non-volatile random access memory (shown in detail in FIG. **6**), for example consisting of electrically erasable programmable read only memory (EEPROM) or flash memory. The non-volatile RAM stores the operating system for the set-top device **100**. The operating system defines the basic functionality of the set-top **100**. For example, the operating system controls how the microprocessor of the DET **102** interprets application programs. The operating system includes the various driver routines permitting the microprocessor to operate the other elements of the set-top **100**. The operating system also includes the basic or 'resident' application under which the DET operates when not running a downloaded application. The resident application preferably emulates a cable television type program reception type user interface for the particular network to which the set-top **100** connects.

One item stored in the non-volatile memory is a channel identifier for a network program channel that will carry the operating system software, for example channel **0**. Typically, an installer will program this value in the DET memory as part of the initial installation procedure, using the keypad on the DET or the remote controller (not shown).

A party providing the operating system upgrade service operates a data carousel application. With this type of application, a digital data stream cyclically repeats, and in accord with the present invention, the network carries the repeating data stream on a broadcast channel. The data stream may include video, audio, data and executable code. For an operating system download, the repeating data consists of a data file containing new operating system code.

The party selling the set-tops to the video information users (VIUs) will provide the operating system updates for those set-top terminal devices. For example, if one of the VIPs sells set-top devices, then that VIP would offer the operating system update service for those set-top devices. If the video dial tone network operator sells the set-top devices, then that operator would offer operating system updates. If the network operator does not operate its own encoder system **11** for other purposes, the operator can make arrangements with one of the VIPs to supply the operating system data carousel through that VIP's encoder system.

To provide the broadcast downloading, a VIP operates a software server, such as server **12**. Typically, the server **12** is a personal computer or the like which compiles the code and/or data for transmission. For applications, such as for controlling navigation through the VIP's program services, the computer compiles application software and data to be processed by that application software. For the operating system upgrade service, the computer compiles a data file containing the instructions which form the various modules of the operating system. The computer cyclically outputs the relevant data in sequence. For the operating system download, the computer repeatedly sequentially outputs the contents of the data file.

The server outputs the data file to the encoder system **14**. The encoder system processes the data and supplies the processed data to the network **15** for broadcast along with

the encoded program information offered by the VIP from the source system **11**. When necessary, the set-top selects the appropriate channel, e.g. channel **0**, decodes the data from the broadcast through the network and recaptures the operating system data file. In the preferred embodiment, the NIM **101** performs the channel selection and conversion back to a data transport stream (e.g. MPEG packets) from the physical layer protocol utilized on the network (e.g. ATM). The DET **102** in turn processes the transport stream to capture the data file. The DET **102** then utilizes the data file to upgrade its stored operating system software.

In the preferred implementation, the channel **0** that carries the operating system upgrade files also carries network program guide information. The network offers 6 Mbits/s channels. In one example, channel **0** will carry the operating system data file at 1.5 Mbits and carry the video and audio packetized elementary streams for the program guide service at a combined rate of 4.5 Mbits/s in a time division multiplexed transport stream at a combined rate of 6 Mbits/s.

The DET operating system upgrade of the present invention can be initiated either automatically or manually. The DET **102** may automatically check the time or number of power-off cycles since the last upgrade, to trigger an operating system upgrade routine. Alternatively, the user may execute a specified sequence of keystrokes on the remote control to call up a menu. One option on the menu is operating system upgrade. Manual selection of the operating system upgrade feature from the menu would trigger execution of the software upgrade routine.

Once initiated, the only difference in the two procedures is whether the DET **102** provides on screen displays during the upgrade procedure. During a manually initiated procedure, the DET **102** will output some form of 'Please Wait' message for display on the screen of the associated television set **103**. During an upgrade procedure automatically initiated, e.g. after a power-off input by the user, the DET will not generate any messages.

When initiated, the DET **102** executes a normal channel selection appropriate to the particular network to receive the channel carrying the software broadcasts. In the present example, the DET **102** instructs the NIM **101** to select channel **0**, and the NIM alone or through interaction with network elements selects that broadcast channel, captures the transport stream therefrom and passes that stream to the digital signal processing circuitry within the DET **102**.

One of the non-writable sections of the memory within the DET stores an operating system upgrade routine. This routine may be stored in ROM or in a sector (e.g. sector **0**) of a flash memory. Once the DET has selected and is receiving channel **0**, the DET microprocessor calls and executes the upgrade routine from memory. The upgrade routine includes information and instructions necessary to extract the operating system information from the MPEG data stream.

The microprocessor of the DET **102** will check the operating system version number carried on the network for the particular type set-top terminal, by comparing data contained in one of the packets from the received transport stream to data stored in memory. If the version number for the operating system broadcast on the network is the same as the version number of the operating system currently running in the DET **102**, then the DET terminates the upgrade process.

However, if the version number for the operating system broadcast on the network differs from the version number of the operating system currently running in the DET **102**, then the DET proceeds with the upgrade process. Specifically, the DET extracts the broadcast operating system from the transport stream from the selected channel and stores that new version in RAM. When extraction is complete, the microprocessor checks and confirms that the extracted and stored version is error free. If no errors are found, the microprocessor transfers the version of the operating system from RAM to non-volatile memory, effectively writing the new version over the old version in the non-volatile memory. The microprocessor checks for errors in the version now loaded to non-volatile memory, and if error free, the microprocessor reboots to begin running under the new operating system.

FIG. **2** shows the elements of the source system **11** in more detail. As shown, the source system includes six sources (13_1 to 13_6) of baseband audio/video information, e.g. in NTSC signal format. The encoder system **14** includes a corresponding number of real time encoders (RTEs) 25_1 to 25_6. Each RTE converts one baseband program signal into digitized and compressed form in accord with the selected protocol. The RTEs supply encoded information to an ATM multiplexer **29**, either directly as shown for live or other real-time type broadcast services or through some form of storage device or server (not shown) for other types of broadcast and IMTV services.

The encoder system also includes a data module **27**. The data module **27** receives the cyclic data output from the software server **12** via an appropriate data interface, e.g. via an Ethernet. The data module **27** formats the data in the same type of packets as produced by the real time encoders 25_1 to 25_6. Preferably, the data module **27** also constructs and inserts certain packets carrying information that the set-tops **100** need in order to find and decode copies of operating systems carried in the packet stream. Because the output from the software server **12** cyclically repeats, the resulting sequence of packets output from the data module **27** also repeats. In an alternate embodiment, the server and data module could be combined, so that the operating system software is stored in memory in MPEG packet form and cyclically, repeatedly output. The data module **27** supplies the packets to another input of the ATM mux **29**. The ATM mux adapts the packets from module **27** into ATM cells in the same manner as for packets from the real time encoders 25_1 to 25_6 and multiplexes the resultant cells into the output stream together with the cells carrying the encoded program information.

In the preferred embodiments, the program material represents a television type program or the like in NTSC format. The video information, accompanying audio information and certain related data are encoded using a standardized digitization and compression technique, such as DIGICIPHER™ or preferably MPEG (moving pictures expert group). Typically, these digital compression protocols also specify a standard packet data format.

In the preferred implementation, the RTEs 25_1 to 25_6 and the data module **27** operate in accord with MPEG II. A detailed discussion of the standard may be found in International Organisation for Standardization Organisation Interationale de Normalisation, "Coding of Moving Pictures and Associated Audio", ISO/IEC JTC/SC29/WG11, CD ISO/IEC 1-13818, February 1994, and a brief summary of MPEG II processing follows.

MPEG is a bi-directional predictive coding compression system, utilizing discrete cosine transformation (DCT) processing to digitize and compress video information. For video information, the encoder will develop reference (I) frames, predictive (P) frames and delta (B) frames.

The number of frames to be coded for each I frame is set in the standardized MPEG syntax, e.g. one reference frame for each group of fifteen frames, or every half second. A prediction is made of the composition of a video frame, termed a P frame, to be located a specific number of frames forward and before the next reference (I) frame, this specific number also is set in the MPEG syntax. Information from previous video frames as well as later video frames is used in formulating the prediction. "Delta" or "B frame" information is developed for coding the video frames between the actual and predicted frames, also by looking at frames in both directions. Rather than updating a whole frame, only the changed (or delta) information is provided for the delta video frames. Typically, between I frames, the frame sequence consists of a repetitive succession of two B frames followed by one P frame.

MPEG II also specifies digitizing and compressing techniques for accompanying audio information. The MPEG II standard provides a standardized format for packetizing the compressed audio and video information and for other data. Under the MPEG II standard, incoming individual video signals and related audio signals are encoded and packetized into respective Video and Audio Packetized Elementary Streams (PES). The video and audio PES's from one or more sources of video programming may be combined with similarly packetized data into a transport stream for transmission or storage.

Each frame of compressed audio or video program information is broken down into a series of transport packets. Data, e.g. in Ethernet protocol form, is also repacketized into MPEG II transport packets. Although the frames can vary in length, e.g. between a full reference I-frame and a delta B-frame, the transport packets have a fixed 188 byte size. Thus, different frames are broken down into different numbers of MPEG transport packets. For example, in a 6 Mbits/s encoding system, a group of frames consisting of a total of 15 frames for one-half second of video (one I frame and a number of P and B frames), breaks down into approximately 4000 transport packets.

The MPEG II standard also permits transport of private or user data as payload information in the 188 byte packets. As discussed in more detail below, each packet includes a packet identifier (PID) value, and the encoder or data module inserts the assigned PID into the packet as part of the packet formatting process. Different PID values are assigned to different programs and content. For example, one program may have a first PID for video, a second PID for audio and a third PID for related data (e.g. closed captioning). The same stream may also contain private data not directly related to the program, e.g. application or operating system software, and a different PID is assigned to packets transporting that data.

As shown in FIG. 3, each 188 byte transport stream packet consists of two or three sections, a 4 byte packet header section, a payload section and/or an optional adaptation field. The header information includes, inter alia, a synchronization byte, a variety of different flags used in reconstruction of the frames, and a thirteen bit program identification (PID) number. PID value 0 is reserved as an indication that the packet includes program association table data (mapping program numbers (PNs) for individual programs into PID values for program maps for those programs). PID value 1 is reserved for identification of packets containing conditional access data, such as encryption information. Other program identification numbers are utilized to identify transport packets with the program source from which they originate.

Periodically, the transport packet for each audio/video program will also include a program reference clock (PRC) value within the optional adaptation field. In a typical 6 Mbits/s MPEG II encoding system, the PRC is present in approximately 10 out of every 4000 video transport packets.

When included, the optional adaptation field includes a section for miscellaneous flags, such as discontinuity counter, private data flag, etc. One of the possible flags carried in this portion of the adaptation field is a program clock reference (PCR) flag. The adaptation field (AF) also includes a section designated for AF options. One of the options this section may carry is the PCR value.

On decompression, the decoder in the DET **102** in sequence reconstructs the frames for a particular program from packets bearing the appropriate PID values, uses the reference frame to form the prediction frames, and then uses the prediction frames and delta information to construct full frames from the delta frames. As discussed in more detail below, circuitry within the DET **102** routes the private data, such as the software download data, to the microprocessor of the DET for further processing.

Returning to FIG. **2**, the data module **27** receives a data stream, e.g. RS-232 or Ethernet, from the software server **12** and converts the data stream to an MPEG II transport stream consisting of packets of the type shown in FIG. **3**. Essentially, the data module **27** subdivides the input data into units which will fit in the payload of MPEG II packets and combines those units with appropriate MPEG II headers to form the MPEG II packets. The information in the added headers identifies the packets containing the software and identifies the payload information as private data.

The data module **27** also inserts one or more appropriate PID values into the packet headers. For example, one PID value would identify the operating system for a first model of DET, another PID value would identify the operating system for a second model of DET, etc. Separate PID values would identify any application software to be broadcast on the same channel.

The data module **27** also constructs a number of packets used to find and decode desired sequences of packets in the stream, for example a program association map (PID 0), one or more program map tables and a network table. The information contained in the map and tables are discussed in more detail below.

The preferred network embodiments utilize ATM transport, therefore the encoder system **14** includes an ATM multiplexer (mux) **29**. The data module **27** receives a repeating or cyclical sequence of one or more data files from the server **12** and supplies a repeating sequence of MPEG II packets to the ATM multiplexer **29**.

In ATM, transfer is asynchronous in the sense that the recurrence of cells that contain information from any particular sender is not necessarily periodic. Each device using an ATM network submits a cell for transfer when they have a cell to send, not when they have an assigned or available transmission time slot. However, the ATM cells may ride in synchronous slots on a high-speed time division multiplexed media, such as a SONET optical fiber. ATM allows any arbitrary information transfer rate up to the maximum supported by the ATM network, simply by transmitting cells more often as more bandwidth is needed.

In ATM, information is organized into cells having a fixed length and format. Each cell includes a header, primarily for identifying cells relating to the same virtual connection, and an information field or "payload". Under presently existing ATM standards, a 53 byte ATM cell includes a cell header

consisting of 5 bytes and a payload consisting of 48 bytes of payload data (see FIG. **4**). The ATM cell header information includes a virtual path identifier (VPI) and a virtual circuit identifier (VCI) to identify the particular communication to which each cell relates. The specific format of the ATM cell is described. for example, in the ATM User Network Interface Specification, Version 3.0, published by The ATM Forum, Mountain View, Calif., also published by Prentice Hall, the disclosure of which is incorporated in its entirety by reference.

FIG. **4** depicts a typical ATM cell format. The ATM cell includes a header section and a payload section. The first 8-bit byte of the header section includes a 4-bit GFC word which provides access control. The first byte of the header section also includes the lower four bits of an 8-bit virtual-path identifier (VPI). The second byte of the header section includes the upper four bits of the VPI and the first four bits of a 16-bit virtual circuit identifier (VCI). The third byte includes the next eight bits of the VCI. The fourth byte of the header section includes the last four bits of the VCI; a 3-bit payload type indicator (PT); and a cell loss priority bit (CLP). The fifth byte of the header section includes an 8-bit header error check (HEC) word. Bytes **6** to **53** carry information and form the ATM cell payload section.

As used here, the ATM multiplexer **29** performs an ATM adaptation function which converts the input information (in MPEG II transport packets) into ATM cells. The ATM multiplexer **29** also performs a multiplexing function to combine cells streams carrying payload data from a number of sources into one higher rate bit stream.

In ATM based networks of the type under consideration here, the MPEG II bit streams are converted into cellular payload data, and cell headers are added. A number of techniques can be used to adapt the transport packets into ATM cells, and certain preferred techniques are described below by way of example.

As noted above, each MPEG packet consists of 188 bytes, whereas each ATM cell includes 48 bytes of payload data. The ATM multiplexer which map the MPEG packets into ATM cells preferably uses two different adaptations to encapsulate MPEG II packets in ATM cells. The first adaptation maps one 188 byte MPEG packet into five ATM 48 byte cell payloads (FIG. **5**A). The second adaptation maps two 188 byte MPEG packets into eight ATE 48 byte cells payloads (FIG. **5**B).

MPEG packets of 188 bytes map efficiently into ATM cells if pairs of packets are mapped into 8 cells. However, a delay is imposed on mapping of a first cell while waiting for the second cell in the pair. To minimize jitter at the decoder, the packets carrying the PCR values need to be encoded and transported quickly. To avoid delaying first packets containing a PCR while processing a second packet, the present system maps first packets containing a PCR immediately, using the five cell adaptation procedure. In a typical video transmission, the PCR is present in approximately 10 out of every 4000 MPEG II packets. Also, at least some of those 10 packets will arrive as the second packet of a pair. Consequently, only a very small number of packets are mapped using the less efficient 5-cell adaptation.

As shown in the simplified block diagram of FIG. **2**, each MPEG type real time encoder RTE **25** supplies a stream of MPEG II packets to the ATM multiplexer **29**. The ATM multiplexer **29** checks the flags in the adaption field (if any) in the first packet to determine if that packet includes a program clock reference (PCR) value. The ATM multiplexer applies the 5 cell adaptation to first packets containing a program clock reference (PCR) value. The ATM multiplexer applies the 8 cell adaptation to pairs of cells wherein the first packet does not contain a program clock reference (PCR) value. Packets containing private data, such as applications and operating system software, will not contain a PRC flag.

For each type of adaptation, the ATM multiplexer **53** will first convert the source packet or pair of packets into a single ATM adaptation layer **5** (AAL5) packet. As part of this conversion, the mux will add an AAL5 trailer, either at the end of the single packet or at the end of the pair of packets. The actual trailer consists of 8 bytes of data, including 4 bytes of cyclic redundancy check (CRC) data, user information (e.g. length), etc.

For a 5 cell adaptation (FIG. **5**A), the AAL5 packet consists of a single MPEG packet of 188 bytes and an 8 byte AAL5 trailer, for a total of 196 bytes. To map this packet into ATM cells, the AAL5 packet is also padded with 44 bytes after the trailer, for a total of 240 bytes of payload data. The ATM mux **53** breaks the AAL5 packet (240 bytes) down into five 48-byte payloads (SAR-PDU) and attaches appropriate 5 byte headers to each payload to thereby form five 53-byte ATM cells.

The header of all five of the ATM cells will contain the VPI/VCI value assigned to the particular communication. For example, for the broadcast service combined with the software downloading, the assigned VPI and VCI value would correspond to network logical channel **0**. For the video and audio portion of the program guide service, the packets would periodically contain a PCR value and periodically would go through the 5 cell adaptation in the normal manner. The header of the first of the five cells also has a bit designated "AAU" which has a value of "**0**" to identify that cell as the first cell. The header of the fifth cell will have an AAU bit value of "**1**" to identify that cell as the last cell.

For an 8 cell adaptation, the AAL5 packet consists of two MPEG packets of 188 bytes and an 8 byte AAL5 trailer, for a total of 384 bytes. The ATM mux **53** breaks the AAL5 packet (384 bytes) down into eight 48-byte payloads and attaches appropriate 5 byte headers to each payload to thereby form eight 53-byte ATM cells. The AAL5 layer is omitted from FIG. **5**B for simplicity. That drawing shows the mapping of two MPEG packets into eight ATM cells with the inclusion of the AAL5 trailer in the last cell.

The header of all eight of the ATM cells will contain the VPI/VCI value assigned to the particular communication. Continuing the above example, if the MPEG data relates to the program guide or the operating system downloading service, the assigned VPI and VCI values would identify logical network channel **0** as in the above discussed example of the five-cell adaptation. The header of the first of the eight cells will have an AAU bit value of "0" to identify that cell as the first cell. The header of the eighth cell will have an AAU bit value of "1" to identify that cell as the last cell.

As noted above, each cell of a particular stream will have a header which contains a virtual path identifier/virtual circuit identifier (VPI/VCI) to identify the virtual circuit that the cells pertain to. All MPEG packets for a given program, whether video, audio or data, will be mapped into ATM cells having the same VPI/VCI. Conversely, cells having a given VPI/VCI will contain data corresponding to only one identified program. Thus, in the above broadcast example, the cells from the one broadcast program all contain the same VPI/VCI value whether the five-cell adaptation was used or the eight-cell adaptation was used.

In the presently preferred embodiment, the ATM mux **29** processes MPEG II packet streams for a combined program

or transport stream capacity of approximately 36 Mbits/s. For simplicity, it is assumed that normal video programs utilize a 6 Mbits/s encoding. The program guide service, however, includes relatively little motion and can be efficiently encoded at a 4.5 Mbits/s rate. The data module therefore can cyclically output the software at 1.5 Mbits/s. The ATM mux **29** therefore receives packet streams from up to six real time encoders (RTEs) **25** and one data module. In a source system **11** offering no software downloading service there would be no server **12** or data module **27**, and the mux **29** would receive six 6 Mbits/s MPEG II streams from the six RTEs. The ATM mux **29** performs the AAL5 adaptations of FIGS. **5**A and **5**B on all of the inputs from the real time encoders 25_1 to 25_6 and the data module **27** (if included). The ATM mux **29** forms the actual ATM cells with assigned VPI/VCI values in the cell headers and combines the ATM cells from all of the programs and the software transmission into a single DS3 bit stream.

In mapping cells from multiple programs to ATM cells and combining cell streams into a signal bit stream, it is necessary for the mux **29** to map the PID value from each MPEG II packet into the correct VPI/VCI value for the corresponding program. The ATM mux **29** therefore is programmed to recognize the PID values of packets for each program and apply the adaptation techniques discussed above relative to FIGS. **5**A and **5**B and to map the PID values into the assigned VPI/VCI values.

At the network node which terminates the ATM cell transport, a receiver captures each ATM cell having a specified VPI/VCI. In the preferred embodiment, the network **15** transports ATM cells through to the set-top terminals **100**, therefore the receiving node would be the subscriber's terminal or set-top **100**.

The element of the network terminating ATM transport will include an ATM demultiplexer (not shown). In the preferred embodiment utilizing ATM cell transport to the set-top terminal devices **100**, the ATM demultiplexer is an element of the NIM **101**. In other network implementations, the ATM demultiplexer may simply reconstruct the MPEG transport streams and supply those streams to some other mechanism for broadcasting the MPEG streams to the set-top devices **100**.

Wherever implemented, the ATM demultiplexer receives a multiplexed ATM cell stream carrying ATM cells relating to a number of programs or sessions. The ATM demultiplexer performs two functions, demultiplexing the combined stream to recover cells relating to at least one communication and ATM to MPEG reverse adaptation to strip off the ATM cell headers and reconstruct the MPEG packets. In the preferred embodiment wherein the ATM demultiplexer is an element of the NIM **101**, as part of the demultiplexing function, the demultiplexer captures all MPEG II packets carried in cells having a single specified VPI/VCI value and provides those packets to a decoder in the DET **102**.

Other demultiplexing functions are possible depending on where the demultiplexer fits into the overall network architecture. For example, the demultiplexer could provide multiple outputs to multiple decoders. For example, the hybrid fiber coax based system disclosed in FIG. **4** of the above-cited 08/304,174 application, an ATM packet handler performs the ATM demultiplexer function. That packet handler provides multiple output rails each of which carries a combined MPEG II packet stream for 4 programs for broadcast in one 6 MHz RF channel. The NIM captures a combined stream from an RF channel, and an MPEG decoder in the DET processes packets for one of the 4 programs based on PID value recognition.

As part of the reverse adaptation functionality, the demultiplexer buffers cells until it finds a cell having an AAU value of "0" in its header (first cell) and another cell having an AAU value of "1" in its header (last cell). The demultiplexer counts the number of cells from first to last to determine the type of adaptation used to map cells.

If the demultiplexer has captured five cells, the demultiplexer pulls out the payload data and uses the CRC data do check for errors. If there are no errors, the original MPEG packet is reconstructed from the appropriate bytes of payload data from the first four cells. Similarly, if the demultiplexer has captured eight cells, the demultiplexer pulls out the payload data, does the CRC based error check, and if there are no errors, the original pair of MPEG packets is reconstructed from the appropriate bytes of payload data from the eight cells.

The DET **102** processes the MPEG II packets in the resultant stream based on their respective PID values. Packets having PID values assigned to audio or video are processed by corresponding decoders and associated driver circuits to produce signals for driving the television set **103** to display the program information to the user. Downloaded software, however, is transferred as private data to the microprocessor of the DET. Of particular note for purposes of the present invention, if the software relates to an operating system, the microprocessor executes the upgrade routine to replace the existing operating system stored in non-volatile RAM with the newly received operating system software.

To facilitate an understanding of the operating system download feature it is useful to consider the structure of the set-top terminal **100** in more detail. A preferred network implementation is discussed below with regard to FIGS. **8**A and **8**B, and a preferred procedure for operating system upgrades executed by the DET **102** is discussed below with regard to the flow chart of FIG. **9**.

The set-top terminal **100** shown in FIG. **6** will connect to a number of different types of digital networks, offering broadcast and point-to-point type services, such as disclosed in commonly assigned application serial no. 08/413,810 filed Mar. 28, 1995 entitled "Access Subnetwork Controller for Video Dial Tone Networks" (attorney docket no. 680-093B), the disclosure of which is incorporated herein entirely by reference. A specific preferred network embodiment is discussed in detail below with regard to FIGS. **8**A and **8**B.

For each different type of network, the terminal **100** includes a network interface module **101** providing the actual physical connection to the particular type of network. For example, in a fiber to the home network, the module **101** would include means for two-way conversion between electrical and optical signals and connections to one or more optical fibers for the necessary two-way transmission. However, the network interface module might be modified for a non-physical communication link, for example, via satellite-to-antenna, especially in rural areas. In the preferred network discussed below, the NIM **101** provides the connection to the coaxial cable type drop.

The network interface module **101** will also perform any format conversion necessary between signal formats utilized by the network and signal formats used within the DET **100**. For example, in the switched digital video type network disclosed below with regard to FIGS. **8**A and **8**B, the network interface module **101** will include means to receive and process a baseband 180 Mbits/s broadband data stream, select a DS-3 from that stream, and process and convert a

selected ATM cell stream into MPEG II bit stream for further processing by the DET **102**.

The network interface module also provides two-way signal conversion and formatting for control signalling between the DET and NIM and for a control signaling channel through the particular network. For example, the network interface module would include means to multiplex and demultiplex signals for transmission/reception over a coaxial cable or optical fiber.

In the illustrated embodiment, the network interface module **101** presents two connections to the DET **102**, a high bit rate broadband connection and a low bit rate signaling connection. The broadband connection is a one-way downstream only connection, but the low-bit rate signaling connection is a two-way connection.

The network interface module **101** takes the form of a plug in module. In one embodiment, the module **101** would be similar to a daughter board or option card which can be plugged into a back plane of a personal computer (PC). In such an embodiment, typically a technician could replace the module in either the field or the shop, to modify a set-top device **100** to connect to and communicate over a different network, and the technician would modify associated communications control software in the system memory. Alternative implementations may use a user replaceable cartridge type network interface module, similar to a video game cartridge, which may include memory in the module for storage of the communications control. As a further alternative, the network interface module could include a digital signal processor controlled by the CPU of the DET **102** and input/output connections compatible with all of the digital broadband networks currently available. The downloaded operating system software stored in the system memory of the DET would control operations of the digital signal processor to send and receive signals in accord with the particular network to which the subscriber chooses to connect the set-top device **100**.

The DET **102** includes a CPU **105**, comprising a 386, 486 PENTIUM™, or Motorola 6800 Series microprocessor **110** and associated system memory **120**. The system memory **120** includes at least 2 mbytes of volatile dynamic random access memory (RAM) **122** and 1 mbyte of non-volatile random access memory (NVRAM) **121**. In the preferred embodiment, the NVRAM **121** is a flash memory device. The CPU **105** also includes a read only memory (ROM) **115**, either as a separate element connected to the microprocessor **110** as shown or as an element within the microprocessor **110**. The ROM **115** stores "loader" programming needed to control wake-up. The non-volatile RAM **121** stores the operating system for the microprocessor **110**. In operation, the volatile RAM **122** temporarily stores applications programs for execution by the microprocessor **110** as well as related data files, and during operating system download operations, the RAM **122** temporarily stores the new operating system.

In the preferred embodiment, the operating system for the DET **102** includes a version of a PC type operating system, e.g. OS-9. In addition, the operating system for the DET **102** includes the various drivers necessary for the DET microprocessor **110** to operate the associated peripherals, e.g. the Digital Audio/Video Processor **125**, the Personal Computer Memory Card Industry Association (PCMCIA) port **155**, the RS-232 transceiver **151**, etc. The set-top operating system also includes the resident cable television emulation software, i.e. as needed to facilitate reception of broadcast programs through the particular network. This operating system is stored in a portion of the non-volatile RAM **121** having a relatively low level of protection. When a new operating system is installed, as discussed more fully below, the new operating system replaces the entire operating system previously stored in the non-volatile RAM. The level of protection here provided enables rewriting the operating system using a broadcast channel download procedure, however, there is sufficient protection to limit storage to only acceptable software from an authorized provider.

A digital audio/video (A/V) signal processor **125**, controlled by the CPU **105**, produces digital uncompressed audio and video signals from the audio and video MPEG encoded packets received from the network through the interface module **101**. The audio/video processor **125** includes an MPEG system demultiplexer **127**, an MPEG video decoder **129**, an MPEG audio decoder **131**, a graphics overlay controller **133** and at least two frames of video RAM **135**.

The MPEG system demultiplexer circuitry **127** recognizes packets in the MPEG data stream received over the broadband channel through the network interface module **101** and routes the packets to the appropriate components of the DET **102** based on the PID values of the respective packets. For example, the MPEG system demultiplexer **127** circuitry recognizes audio and video packets in the MPEG II data stream and routes those packets to the decoders **129**, **131**, respectively. The MPEG system demultiplexer **127** routes private data, such as downloaded software, to the microprocessor **110**.

The MPEG video decoder **129** decompresses received video packet signals to produce a digital video signal, and the MPEG audio decoder **131** decompresses received audio packets to produce left and right digitized stereo signals. For at least some functions, the MPEG decoders **129**, **131** may be controlled in response to signals from the microprocessor **110**. The MPEG video decoder **129** will internally include at least two frames (e.g. 8 mbytes) of RAM (not separately shown) for use as a frame reorder buffer during the MPEG video decoding process, and the MPEG audio decoder **131** also may include some buffer memory.

The video RAM **135** is not a specialized "video RAM" as that term is sometimes used in the television art. The RAM **135** is actually a standard digital data RAM, of appropriate size, which is used in the DET to store digitized frames of video data. The RAM within the MPEG video decoder **129** likewise consists of standard digital data RAM.

The graphics display generator produces displays of text and graphics data, such as a selection menu received over the signaling channel, in response to instructions from the CPU **105**. The video RAM **135** sequentially receives each frame of digitized, uncompressed video information, as output from the MPEG video decoder **129**. The video RAM **135** also receives digital information and read/write control signals from the graphics overlay controller **133** representing the several planes of text and graphics information and combines that information with the frames of decompressed video to produce composite video frames.

The graphics overlay controller **133** and the video RAM **135** actually cooperate to manipulate five different planes of video information, four of which can be active at any one time, to produce the composite video frame output signals. The individual planes comprise the decoded MPEG video frames, a cursor, two graphics/text image planes manipulated by the microprocessor **110** and a backdrop plane. The backdrop plane would be switched in to replace the plane representing the decoded MPEG video frames, e.g. to present a blue background instead of the MPEG video background.

When there are no graphics or text, the composite frames would correspond entirely to the uncompressed received video frames output by the MPEG video decoder **129**. When no received video frames are to be output, either when none are received or when they are to be entirely replaced, the information from the graphics overlay generator **133** would specify a background and the active planes of text or graphic information. When received video frames are combined with text and/or graphics, the composite video frames include the uncompressed received video frames with selected pixels thereof replaced with graphics or textual data display pixels specified by the graphics overly controller **133**. In this last situation, the graphics overlay controller would deactivate the backdrop plane.

Under certain circumstances, the video RAM **135** also serves to freeze video frames. For example, when a video transmission ends for some reason, the RAM **135** will contain the video and associated graphics information for the frame last received and displayed. The DET **102** can continue to output this frame as a still video output signal for some period of time.

The DET **102** also includes audio and video digital to analog converters and appropriate drivers to produce output signals compatible with a conventional television set. Specifically, the converter and driver circuitry of the DET **100** includes audio digital to analog converters (DAC's) 134_L, 134_R, an audio mixer **136**, an NTSC encoder **137**, and an RF modulator **139**.

The DAC's 134_L and 134_R receive the uncompressed left and right digitized audio signals output by the MPEG audio decoder **131**. In response, the DAC's 134_L and 134_R produce baseband analog audio signals for output to individual baseband output terminals. The audio mixer **136** also receives the baseband audio signals from the DAC's 134_L and 134_R. The mixer **136** combines the left and right analog audio signals to produce a monaural audio signal as the audio input to RF modulator **139**.

The NTSC encoder **137** also performs a digital to analog converter (DAC) function. In response to the digitized video output signals from the video RAM **135**, the NTSC encoder **137** produces a baseband analog video signal in standard NTSC format. The baseband NTSC video signal is supplied to an output terminal of the DET **102**. The baseband NTSC video signal is also supplied to the RF modulator **139**. The RF modulator **139** responds to the mono audio signal, the NTSC video signal and an RF signal from a local RF oscillator **141**, to produce a selected standard RF television signal on an available TV channel, typically channel 3 or channel 4.

The type of connection of the DET **102** to the television set depends on the capabilities of the user's television set. If the user has a monitor type television capable of receiving baseband video and stereo audio inputs, the appropriate terminals of the television would connect directly to the video and audio output terminals of the DET **102**. If the subscriber does not have such a television monitor, then the RF output of the modulator **139** would be connected to the cable or antenna input connection of the television, e.g. by coaxial cable. Alternatively, the digitized video and audio may go to separate output terminals (not shown) for connection to inputs of digital display devices, for example, for high definition television (HDTV) sets.

Each DET **102** also includes means to receive selection signals from a user, and under at least some circumstances, transmit appropriate data signals over a narrowband channel through the particular video network. For example, the DET **102** may send and receive control data through a signaling channel on the subscriber's loop or drop cable. In the preferred embodiment, a switching element of the network routes selected broadcast channels to the set-top **100**. The DET **102** provides selection signals to the NIM **101** for upstream transmission over the signaling channel to that switching element to identify a requested channel. In a similar fashion, the set-top terminal may transmit upstream signaling information through the signaling channel for transport through the network to a video information provider offering interactive services.

In the embodiment illustrated in FIG. **6**, the DET **102** includes an infrared (IR) receiver **145**. The (IR) receiver **145** responds to inputs signals from a user operated IR remote control device (not shown) similar to that used today for controlling televisions and video cassette recorders. In response to the IR signals, the receiver **145** produces corresponding digital data output signals. The microprocessor **110** interprets the digital data signals by the IR receiver **145** as input commands. The precise interpretation of specific command signals can vary based on the downloaded applications programming and/or the operating system software currently stored in the system memory **120**. For example, in response to certain input commands, the microprocessor **110** controls cursor position and alphanumeric information displayed as graphics and text on the associated television set. The microprocessor **110** will also respond to an appropriate input command from the user to formulate a message for upstream transmission though the network interface module **101** and the signaling channel of the particular connected network, e.g. to select a broadcast channel.

The set-top terminal device **100** of the present invention is an open interface device in that it interacts with equipment of a large number of service providers (often referred to as "VIPs") to offer users a wide array of video and interactive multi-media services. In the preferred embodiments, the digital entertainment terminal (DET) **102** is a programmable device to which different individual video information providers (VIPs) can download applications software, and at least one VIP or the network operator (the party selling the set-top device to the end user) can download the operating system software.

In the ROM **155** and/or a relatively high-level write protected portion of the NVRAM **121** (e.g. sector **0** of flash memory), the DET will store a loader program similar to the bios of a PC. The NVRAM **121** will also store an operating system. The loader program and operating system in the ROM and the non-volatile RAM will include sufficient programming to control initial communications and define interfaces and drivers, e.g. for graphics to define the base line functionality of the DET for all service applications the DET will run. This stored software also includes the resident application, which in the preferred embodiment is a CATV-like broadcast program reception routine appropriate for the particular network connected to the set-top terminal **100**. The ROM or the most write-protected portion of the NVRAM also stores an operating system upgrade routine for controlling the DET process of upgrading the operating system through a broadcast channel download operation.

The DET **102** of the present invention may also include a number of additional interface devices. In the example illustrated in FIG. **6**, the DET **102** includes an IR transmitter **147**. The transmitter **147** responds to digital data signals from the microprocessor **110** and outputs corresponding IR signals for wireless transmission. The IR transmitter **147** and IR receiver **145** may operate together to provide a two-way wireless data communication link to some remote device,

such as a personal data assistant (PDA) or pocket organizer. Alternatively, the IR transmitter may send signals to a remote display device for use in a service not requiring the TV set. For example, in an audio on demand service, the IR transmitter would send display data to an LCD display located near the user's stereo system.

The illustrated DET also includes an RS-232 transceiver or interface **151** connected to the microprocessor **110**. An RS-232 port is a standardized two-way serial data interface typically used for connecting computers to peripheral devices, such as modems. In the present system, the RS-232 interface **151** might provide a serial data connection to an external personal computer (PC), such that the DET permits communications between the PC and the broadband network. Alternatively, this port might connect the DET to a printer, e.g. to print coupons during home shopping/browsing services. A hand-held diagnostic terminal might also connect to this port during servicing of the DET. The communications and protocols offered by the DET through the interface **151** would be controlled by the operating system and applications program software downloaded into the system memory **120**.

FIG. **6** also shows the DET **102** including a magnetic card reader **153** connected to the microprocessor **110**. This reader **153** could be used to scan credit card information encoded on magnetic strips on commonly available credit cards. In a home shopping and purchasing service, controlled by the downloaded software, the user would scan their own credit card through the magnetic card reader **153** as part of the payment operations. The reader could also have magnetic write capabilities to perform debit card operations.

The illustrated DET **102** further includes a personal computer memory-card interface adapter (PCMCIA) port **155**. This is a two-way interface for connection to and communication with a flash memory module, such as is now incorporated into advanced "smart card" devices. In a medical service, a user might communicate with a medical information database through the DET **102** and the broadband network. The user's personal medical history information could be read from the smart card and subsequently updated on the smart card, through the PCMCIA port **155**. Another use of this port might involve communication to a connected video game system to download video game software to the video game system and/or play interactive video games. Although specified as a "memory" port and mapped by the CPU as part of its system memory space, the devices connected to this port **155** can have other data processing capabilities, e.g. buffering and modem communication capability. As discussed below, a technician may also use a PCMCIA card to load operating system software into the NVRAM **121**, e.g. when there is a fatal flaw in the currently stored software.

In the current implementation, the PCMCIA port **155** will carry 6 Mbits/s of data, but the port can be designed for higher speeds such as 20 Mbytes/s. Another use of this port would be for connection to an Ethernet card or other Local Area Network (LAN) card to permit data communications between the DET **102** and one or more computers. The set-top **100** would provide the computers with communications services through the broadband network, for example to receive high speed downloads of new or updated software for those computers. Although similar functions are possible through the RS-232 transceiver **151**, the data rate through the PCMCIA port **155** is much higher.

FIG. **7** shows the segments of memory in the DET **102**. The non-volatile portion of the memory consists of the ROM **115** and 1 mbyte of non-volatile RAM **121** as discussed above. Preferably, the non-volatile RAM **121** consists of 1 mbyte of flash memory. The volatile portion of the memory consists of 2 mbytes of DRAM **122**.

The use of flash memory in the preferred embodiment facilitates control of the write operations for different programs stored in the sectors of the memory **121** as a form of memory access control. Access to the different programs is limited by different types of flash memory "LOCKs," each of which requires a predetermined bit pattern to unlock the write operation and permit write access to the memory sectors.

The ROM alone or in combination with a write-protected portion of the NVRAM stores a loader program for controlling many of the wake up functions of the CPU **105**. In the preferred embodiment, the write-protected portion of the NVRAM is sector **0**. To write code to sector **0** requires a bit pattern, corresponding to 'LOCK1' shown in FIG. **7**, which is available only to a technician. The other sectors of the flash memory type NVRAM **121** store the operating system. A bit pattern providing authorized access to write code in the other sectors, corresponding to 'LOCK2' shown in FIG. **7**, will not enable writing of code to sector **0**. As discussed more fully below, the use of these different bit patterns to control the write operation to the sectors of the flash memory effectively limits who may access the various sectors.

The loader program is analogous to a BIOS (basic input/output system) in a PC. The loader program provides initial instructions to the microprocessor **110** to carry out a series of hardware diagnostics during an initial boot routine and to boot up the operating system stored in the NVRAM **121**. If faults are detected, the loader routine will cause display of error codes and instructions on the associated television screen. For example, if the loader routine results in detection of a network error, the DET might generate a display instructing the subscriber to call a telephone number assigned to the network operating company together with a four digit code indicating the type of network fault, e.g. lack of a signaling channel or lack of a broadband channel. Alternatively, if faults in the DET **102** are detected, the display would instruct the subscriber to call a telephone number assigned to the DET vendor together with a four digit code indicating the type of DET equipment fault, e.g. operating system memory error.

In the presently preferred embodiment, the ROM **115** or sector **0** of the flash memory type NVRAM **121** also stores an operating system upgrade routine. The upgrade routine includes information and instructions necessary to extract the operating system information from the broadcast MPEG data stream, in accord with the present invention.

As noted, a specified bit pattern is needed to authorize overwriting of certain sectors of a flash memory. The operating system upgrade routine includes the bit pattern needed to authorize overwriting of all sectors except sector **0**. The operating system can also be overwritten by a technician, e.g through the PCMCIA port **155**, and the software for overwriting the application used by the technician would include the bit pattern necessary to overwrite sector **0**.

Occasionally, it may be necessary for a technician to load software into the DET, for example when the DET has failed due to a fault in the operating system software stored in the NVRAM **121**. In practice, the technician turns the set-top **100** off and inserts the PCMCIA memory card in PCMCIA port **155**. The technician then turns the set-top device **100** back on. In the currently preferred implementation, the

PCMCIA card consists of 2 Mbytes of flash memory and contains the operating full system. The software stored in this card also includes the portion of the loader or boot routine and/or the operating system upgrade routine normally stored in sector **0** of the flash memory of the DET **102**.

The driver program used for the PCMCIA port **155** is similar to that for a floppy disc drive, and the data files on the PCMCIA card are arranged and formatted in a fashion similar to files on a floppy disc. When the DET **102** powers-up, the microprocessor **105** checks the various peripherals in a manner analogous to a power-up boot routine in a personal computer. Of particular note here, the microprocessor **105** checks and determines that a PCMCIA card is present in port **155**. Rather than booting up the operating system in the system memory **120**, the microprocessor detects the bit pattern defining sector access to the DET's internal flash memory in the memory of the PCMCIA card and uses that bit pattern to initiate a sector by sector rewrite of the operating system. Specifically, the operating system stored on the PCMCIA card is transferred sector by sector to the flash memory in the DET **102**. The microprocessor then initiates a reboot of the operating system from the internal flash memory to run the new operating system.

The bit pattern used to initiate a write operation into sector **0** of the NVRAM **121** will be kept secure and not broadcast through the network. A technician may utilize this code as described above, but information service providers offering broadcast programs or offering interactive services through the network will not be aware of this bit pattern and will not have authorization to modify the files stored in sector **0**.

The operating system upgrade routine typically will include the bit pattern corresponding to LOCK2. The party who sold the DET, e.g. the network operator or a video information provider (VIP), will broadcast software and associated control messages to trigger the actual upgrade of the operating system, and the stored upgrade routine will provide the bit pattern needed to actually write the new operating system software to the relevant sectors of the NVRAM **121**.

The 2 MbyteRAM **122** serves principally as a scratch pad memory and as storage for applications software. The application software may be downloaded to the DET **102** through the broadcast channel in essentially the same manner as the operating system from the same provider offering the operating system upgrade service, or any other VIP may download application software through another broadcast channel or through a point-to-point link used for interactive multimedia (IMTV) services.

When the set-top **100** is first connected to power the microprocessor **110** runs its self-diagnostics as specified by the loader routine and boots up the operating system stored in the NVRAM **121**. The DET **102** does not start executing any of the processes, such as MPEG demultiplexing and decoding, because it is not yet in an 'ON' state. The DET instead runs in a lower power state waiting for an ON command from the user. When the user turns the box **100** 'ON', the microprocessor **105** is already up and running and can quickly initiate channel selection and display. When turned 'OFF' after use, the DET returns to the low power standby state, and the HDT **230** will retain last channel viewed data in memory for use as the initial channel selection when the user next turns the DET ON. As part of the return to the low power state, the microprocessor **105** may repeat the self-diagnostics procedure and reboot the operating system.

When the user turns the box 'off', the microprocessor turns off the various output drivers thereby terminating the television display functionality. To the user, the set-top **100** appears off. The microprocessor **105**, however, remains powered and as part of its self-diagnostic routine can determine whether or not an operating system upgrade is necessary, as discussed in more detail below. In the preferred network embodiment, when the user turns the box **100** 'ON', the microprocessor **110** initiates a channel selection of the last viewed channel through interaction with the HDT **230** and begins displaying the information from that channel.

FIGS. **8**A and **8**B together illustrate a preferred architecture for a full service network providing the terminal operating system downloading service in accord with the present invention. The illustrated network provides transport for broadband services including broadcast video and IMTV type services, such as video on demand. The network also provides interactive text services and voice telephone services.

The illustrated network comprises an ATM backbone network and a local loop network. For broadcast services, the ATM backbone network includes an ATM edge device **226** and optical fibers $\mathbf{220}_1$, $\mathbf{220}_2$ from the VIP headends to that edge device. The ATM backbone network includes four optical fibers $\mathbf{229}_1$ to $\mathbf{229}_4$ carrying broadcast programming from the ATM edge device **226** to a large number of host digital terminals (HDT's) **230** (only one of which is shown in FIG. **8**B).

For interactive multimedia television (IMTV) services, the ATM backbone network includes at least one ATM switch **250**. The ATM switch is controlled by a PVC controller **254**. A subscriber wishing to initiate an IMTV session interacts with a level **1** gateway **252** which in turn communicates with the PVC controller **254** to obtain the requisite bandwidth through the switch **250**.

The local loop network consists of the HDT's **230**, two-way optical fiber pairs between the HDT's **230** and optical network units **232** (ONU's), and coaxial cables **240** and twisted wire pairs **234** connecting the ONU's to the subscriber premises equipment. Control processor elements (not shown) within the HDT's **230** and a video administration module (VAM) **242** control the broadcast services.

Signaling communications between the DET's **239** and the serving HDT **230** utilize a consumer electronics (CE) bus protocol. In the presently preferred embodiment, signaling communications between the other nodes of the network ride on an X.25 packet switched data network **246**. In future implementations, instead of data network **246**, the ATM switch **250** will carry the signaling traffic together with the IMTV broadband traffic.

For simplicity and ease of understanding, it is assumed here that the network is set up to transport broadcast services from two VIP's headends. In practice there may be more than two broadcast VIPs on the network together offering at least 384 channels of broadcast programming. Each of the broadcast video headends includes one or more of the source systems **11** discussed above relative to FIGS. **1** and **2**. Each source system **11** digitally encodes up to six audio/video programs in MPEG II format, and at least one of the sources encapsulates the software data carousel containing the operating system information in MPEG II transport packets. As discussed above, each encoder system includes an ATM multiplexer for adapting the MPEG packets into a single ATM stream at a DS3 rate.

In the illustrated example, the DS3 is actually a one-way asynchronous bit stream. The transmission of ATM cells in an asynchronous DS3 signal requires a common clock reference in order to ensure frame alignment between the

ATM multiplexer and a super trunk transmitter **218**. The ATM mux therefore presents the MPEG II packet channels in ATM cell format in accordance with a physical layer convergence protocol (PLCP). The PLCP is a framing structure used to ensure that ATM cells are aligned with respect to a corresponding video frame, even though there may exist drifting of a start and end of a typical DS3 frame. Specifically, the PLCP references a DS3 header and identifies the location of each ATM cell with respect to the DS3 header. Since the DS3 frame contains a maximum of twelve ATM cells, the PLCP notes the location of each of the twelve cells with respect to the DS3 header. Therefore, even though there may be DS3 frame drifting, the PLCP ensures alignment, from a cell perspective, between the cell layer and the DS3 frame so that each of the twelve ATM cells within each DS3 frame can be located.

All broadcast service type video information providers (VIPs) supply programming to the network in the form of a DS3 type, MPEG II encoded ATM streams such as that output by the source systems **11**. The DS3 bit stream from each system **11** goes to one input of a super trunk transmitter (STT) **218**. As discussed below, the STT **218** combines a number of DS3 ATM cell streams into one higher rate bit stream and converts the electrical signals to optical signals for transmission over a trunk fiber **220**. One broadcast VIP may have a number of STT's **218**, and the network will actually carry optical broadcast streams from multiple broadcast service VIPs.

If the optical transmissions are SONET compliant, the super trunk transmitters **218** and super trunk receivers **224** would operate at an OC rate to transport a standard number of DS3 bit streams. For example, OC-12 equipment will transport 12 DS3 bit streams, OC-18 will transport 18 DS3 bit streams, etc. It would also be possible to use an asynchronous optical protocol.

The preferred embodiment uses super trunk transmitters and receivers manufactured by Broadband Technologies, Inc. of Research Triangle Park, N.C. The preferred super trunk transmitters perform a bit stream interleave type multiplexing. The preferred super trunk transmitters (STT's) **218** are capable of receiving DS3 bit streams from up to sixteen sources, for example up to sixteen source systems **11**. Each super trunk transmitter **218** combines those DS3 rate bit streams into a single higher rate bit stream, converts that electrical stream into an optical signal stream and transmits the optical stream over a fiber, such as **220**. The optical fibers **220** each transport up to 16 DS3 streams, wherein each DS3 includes six 6 Mbits/s MPEG II encoded transport streams, for a maximum capacity on the fiber of 96 channels per fiber.

According to the preferred embodiment, the network includes a plurality of parallel trunk fibers **220** from different STT's **218** servicing a variety of VIPs. Each fiber **220** goes to a super trunk receiver **224**. A different set of input broadcast broadband (e.g. television) signals are encoded and multiplexed in a manner similar to that discussed above to produce the combined DS3 bit stream (up to 96 channels) for transport over each respective optical fiber **220**. The network preferably will service up to 50 VIPs.

The trunk fibers **220** are routed to super trunk receivers **224**, each of which recovers up to sixteen DS3 bit streams from the corresponding optical signal stream. Each super trunk receiver **224** supplies each recovered DS3 rate stream to one input node of the ATM edge device **226**. The preferred ATM edge device **226** receives at least sixty-four DS3 inputs from the STRs **224**. The ATM edge device **226** performs policing and grooming on the input ATM cell streams.

The ATM edge device **226** monitors incoming DS3 data streams and determines whether ATM cells within the data streams should be passed to the network or blocked. This functionality of the edge device **226** serves to police incoming cells based on their VPI/VCI values. The edge device will pass cells only if the VPI/VCI values in the cells correspond to a value indicated as valid in the data tables programmed into the edge device. If a cell does not have a valid VPI/VCI value for a currently active program channel, the edge device will not pass that cell to an output port.

Also, the ATM edge device **226** performs policing of DS3 ATM cell streams by monitoring the data rate of incoming data streams from VIPs. For example, if one VIP has subscribed by contract to transmit a particular channel at a data stream rate of 6 Mbits/s, the ATM edge device **226** will prohibit or drop ATM cells having the assigned VPI/VCI value that are transmitted above the subscribed bit rate; in this case, a 6.5 Mbits/s stream would be rejected as an unauthorized rate.

In addition, the ATM edge device **226** provides a grooming function, whereby ATM cells are analyzed, on a cell-by-cell basis, to determine if they should be transmitted on the network. Specifically, each ATM cell having a valid VPI/VCI value is switched through the ATM switch fabric assigned to carry the program identified by the VPI/VCI value. The ATM edge device **226** combines a specified six ATM cell streams into a DS3 bit stream for output on each DS3 output port.

This ATM cell mapping enables DS3 ATM cell streams that are transmitted at less-than-full capacity on the trunk fibers **220** to be mapped onto output DS3 streams operating at full capacity. Although each optical fiber **220** has a capacity of transporting up to 16 fully loaded DS3 ATM streams, at least one optical fiber **220** from two or more VIPs typically will not be operated at capacity, when broadcast VIPs do not offer an even multiple of six channels or when VIPs using the optical fibers have varying bandwidth requirements over time. The ATM edge device **226** processes all incoming DS3 bit streams from all of the optical fibers **220**, and maps the ATM cell streams from those inputs into at least one and preferably sixty-four condensed, or combined DS3 output bit streams for further transmission through the network. Thus, the ATM edge device is able to fully load the optical fibers $\mathbf{229}_1$ to $\mathbf{229}_4$ serviced by the STT's $\mathbf{228}_1$ to $\mathbf{228}_4$ to fully load the downstream broadcast capacity of the network.

The ATM edge device **226** outputs each groomed DS3 stream to one input of a super trunk transmitter (STT) similar in structure and operation to the STTs **218** discussed above. In a preferred embodiment, each DS3 from the edge device **226** goes to one input of the four STT's $\mathbf{228}_1$ to $\mathbf{228}_4$. The preferred embodiment can carry up to 384 broadcast program channels over a capacity of 64 DS3 signal paths (four fibers $\mathbf{229}_1$ to $\mathbf{229}_4$ each carrying sixteen DS3's in a manner similar to the maximum possible on each fiber **220**). In that embodiment, the STT's $\mathbf{228}_1$ to $\mathbf{228}_4$ receive the 64 DS3 ATM streams from the ATM edge device **226**, and output the ATM streams over four parallel optical fibers. SONET or other protocols could be used on the fibers $\mathbf{229}_1$ to $\mathbf{229}_4$ and/or the network could include additional fibers.

The ATM edge device may comprise a relatively small capacity ATM switch. A more detailed description of the ATM edge device and its operation is disclosed in commonly assigned U.S. patent application Ser. No. 08/380,744 filed Jan. 31, 1995 entitled "Full Service Network Using Asynchronous Transfer Mode Multiplexing" (attorney

docket no. 680-109), the disclosure of which is incorporated herein in its entirety by reference.

The ATM edge device **226** outputs at least one DS3 stream of combined ATM streams and preferably **16** such streams to each of four super trunk transmitters (STT's) 228_1 to 228_4. The ATM edge device **226** will preferably output up to 64 DS3 bit streams to the four STT's 228_1 to 228_4. Each particular STT 228_1 to 228_4 combines the input DS3 bit streams into an optical stream, in a manner similar to that of STTs **218**, for transmission on one of the four fibers 229_1 to 229_4.

The signal stream on each optical fiber 229_1 to 229_4 is applied to a bridge circuit **231** (FIG. **8B**) to supply the optical signal stream through corresponding trunk fibers to a large number of Host Digital Terminals (HDT's) 230 distributed throughout the serving area. The bridge circuitry includes passive bridging elements and may include active bridging elements.

The preferred embodiment utilizes HDTs manufactured by Broadband Technologies, Inc. of Research Triangle Park, N.C. Each HDT **230** handles up to 256 subscribers by switching DS3 bit streams carrying selected program channels onto optical fibers to up to 32 optical network units **232** (ONUs). The ONUs **232** transfer the DS3's over respective coaxial cables to subscriber terminals for display on associated television sets.

The downstream optical fibers **233** from the HDT **230** to each connected ONU preferably transport 24 DS3 ATM cell streams, e.g. using OC-24 SONET compliant equipment. Together, the downstream fiber **233** and upstream fiber **237** also provide transport for 2-way telephone communications and 2-way signaling channels. In an alternate embodiment, the network might use a signal fiber between the HDT and each ONU and provide both downstream traffic and upstream traffic on that one fiber.

In the illustrated embodiment, the downstream optical fiber from the HDT **230** to each ONU **232** may use either an asynchronous optical protocol, or the synchronous (SONET) OC rate transport. The ONU's **232** provide appropriate interfacing between the voice channels on the fibers and twisted wire pair **234** for telephone service into the subscriber premises.

Each video information user (VIU) premises **236** is preassigned three DS3 slots on the downstream fiber from the HDT **230** to the ONU **232** for broadband service. Each home or living unit **236** is preferably allocated a capacity of four set-top terminal devices **100**, with three being active at any one time. A coaxial drop **240** for each premises **236** carries 180 Mbits/s baseband digital data, which will simultaneously transport three 45 Mbits/s DS3 bit streams. For a particular VIU premises **236**, the three DS3 channels on the fiber from the HDT **230** to the ONU **232** and on the drop cable **240** are individually assignable to different set-tops **100** within the subscriber's premises **236**. The ONU **232** performs optical to electrical conversion, separates out the DS3's received over the downstream optical fiber **233** from the HDT **230** and supplies the selected DS3 bit stream to appropriate channels on the coaxial cables **240** going to the respective subscriber premises **236**.

For narrowband signaling information, the ONU **232** passes all of the downstream signaling data received from the HDT **230** on fiber **233** through to all of the coaxial drop cables **240**, so that for signaling data the cables look like a common bus shared by all of the connected set-tops. In the downstream direction, signaling packets are interleaved with the ATM cell stream packets. In the upstream direction, the signaling channel on the coaxial cable **240** is in a different frequency portion of the spectrum from the downstream DS3 transmissions. The ONU combines all of the upstream signaling packets from subscriber drop cables **240** into a digital data stream and transmits that data stream together with digitized upstream telephone service signals over the upstream fiber **237** to the HDT **230**.

Each set-top **100** comprises a DET and NIM, as discussed above. In this embodiment, the NIM connects to the coaxial drop cable **240** to send control signals to the ONU **232** and receive video and data signals from the ONU **232**. The NIM includes means to selectively demodulate received data from an assigned one of the three DS3 slots on the coax cable **240** and an ATM demux for mapping ATM cells back into the corresponding MPEG packets. As discussed above, the DET includes an MPEG II audio/video (A/V) decoder. Specifically, the ATM demux captures and processes ATM cells bearing specified VPI/VCI header information corresponding to the selected program channel from the DS3 stream. The MPEG II decoder in sequence reconstructs the frames for a particular program from packets bearing the appropriate PID value, decompresses the compressed digital video information, and displays the decompressed digital video information in the appropriate format.

Each set-top **100** includes means to receive selection signals from a user via remote control, and as noted above, the set-top responds by transmitting appropriate data signals over a narrowband signaling channel on the coaxial drop cable to the ONU **240**. According to the preferred embodiment, the narrowband signaling channel uses X.25 or a consumer electronics (CE) bus protocol. With the CE bus protocol, for example, the active set-tops **100** are assigned signaling time slots by the HDT **230**, and each active set-top **100** transmits channel selection data upstream to the ONU **232** in its assigned slot.

The ONU **232** multiplexes the data signals from the set-tops it services together and transmits those signals to the HDT **230** over an upstream channel on an optical fiber. If the data represents selection signals, the HDT responds to that data as outlined above, and stores data identifying each subscriber's selections for subsequent periodic uploading to a Video Administration Module (VAM) **242**. The HDT's **230** connect to the VAM through a first X.25 packet data communication network **246**.

The operations of each HDT **230** are controlled by data tables stored within the HDT. The video information providers (VIPs) provision various services for their subscribers by establishing appropriate mapping and profile data in the tables in the HDT's **230**. The VIPs, however, do not have direct access to the data tables within the HDTs. Instead, the VIP's access the VAM **242** through a personal computer interface **244** and the X.25 data communication network **246**. The VIPs **210** provide provisioning data through the operations and support system (OSS) **244** to the VAM **242**, and the VAM **242** periodically downloads that data to the appropriate HDTs **230**.

The provisioning data downloaded to the HDTs **230** includes channel mapping information and subscriber authorization control information. The channel mapping information specifies what programs are carried on each ATM virtual circuit, within each DS3, on each respective optical fiber. The HDT **230** accesses the channel mapping information in response to each program selection by a subscriber to route the correct DS3 to the requesting set-top **100** and to inform the set-top **100** as to which virtual circuit within that DS3 carries the requested program. The authorization con-

trol data indicates which programs each subscriber is authorized to access, e.g. because that party has subscribed to the particular program service and is not delinquent in bill payments. When a subscriber requests a program, the HDT **230** checks this data to determine whether or not to supply the program to the subscriber's set-top **100**.

In operation, the network administration and support personnel enter the VIP profile information including the VPI/VCI assignments in a database (not shown) and manipulate that database to define necessary routing tables for transport of the VIP's broadcast program channels through the network. The database then outputs appropriate information for programming the ATM edge device **226** and information for programming the VAM **242**. The VAM **242** periodically updates the actual control tables in each HDT **230** via communications through the X.25 signaling network **246**. In particular, the data downloaded to the HDTs **230** indicates the DS3's on each fiber. The data downloaded to the HDTs **230** also indicates the VPI/VCI values for each program channel within each DS3. In the preferred embodiment, the data in the HDT will also include an initial PID value used in capturing and decoding the MPEG II packets for each program channel, e.g. the PID value for the program map packet for the particular program.

In operation, each time a subscriber turns on a set-top **100**, the set-top transmits an appropriate signaling message upstream to the HDT **230**. The HDT stores a table of valid equipment ID's for the set-tops it services. The signaling message transmitted to the HDT **230** at turn-on includes the equipment ID for the particular set-top **100**. When the HDT **230** receives the initial signaling message from the set-top, the HDT executes a routine to initialize the set-top. As part of this initialization routine, the HDT **230** validates the set-top equipment ID and assigns one of the DS3 slots on the downstream fiber **233** to the ONU **232** to that set-top for as long as that set-top remains on. Also, one of the DS3's on the subscriber's coaxial drop cable **240** from the ONU **232** is assigned to the set-top **100** for the duration of communications.

At the same time, the HDT **230** will complete a two-way signaling communication link with the DET in the particular set-top **100**. At least on the coaxial cable portion, the packets relating to the signaling link are identified by header information identifying the particular link, i.e. a signaling identifier assigned to this communication between the HDT **230** and the particular set-top **100**. As part of the initialization routine, the HDT **230** sends one or more signaling messages to the DET in set-top **100** identifying the signaling channel assignment and the DS3 assignment. Specifically, for the signaling link, the HDT **230** assigns the next idle signaling ID to this call and informs the DET of that signaling ID assignment.

When a subscriber selects a broadcast program, the subscriber's set-top **100** transmits a channel request message, including the equipment ID of that set-top and channel selection information, upstream through the signaling link to the HDT **230**. Using portions of the stored data tables, the HDT **230** checks to determine if the particular set-top is permitted to access the requested channel. The access decision may relate to whether or not the VIU has subscribed to the program service requested. Alternatively, the HDT **230** may execute a PIN/password routine discussed later to determine if the person currently operating the set-top is allowed access to the particular broadcast program service.

If the subscriber is permitted access to the requested channel, the HDT **230** switches the DS3 bearing the requested channel from one of the trunk fibers onto the DS3 assigned to the requesting set-top **100** on the fiber **233** going to the ONU **232** serving the particular subscriber. The ONU **232** in turn switches the assigned DS3 on the fiber **233** onto the DS3 assigned to the particular set-top **100** on the drop cable **240** into the VIU's premises **236**. The HDT **230** addresses a downstream control message to the set-top using the assigned signaling call ID. The control message identifies the VPI/VCI of the requested program within the DS3 and the MPEG PID value for the MPEG PROGRAM MAP TABLE (PMT), so that the set-top **100** can select and begin decoding MPEG II transport packets for the selected program to produce standard signals for driving a television set **103** or for reception of software.

For example, for an operating system download operation, the VPI/VCI value would identify cells relating to channel **0**, and the PID value would identify the program map for the operating system for the particular type of set-top. The NIM **101** (FIG. **6**) would select cells having the specified VPI/VCI value and adapt those cells back into MPEG packets as discussed above. The NIM **101** hands off the 6 Mbits/s MPEG transport stream for channel **0** to the MPEG system demux **127**. In the preferred embodiment, this stream contains the audio and video information for the program guide service and packets containing data from the operating system file. The DET **102** utilizes the downloaded PID value to capture and process the packet containing the program map table, and uses the data from that table to identify the PID value of packets containing the actual operating system file. The microprocessor **110** uses that PID value to instruct the MPEG demux **127** to capture and forward operating system data packets to the microprocessor **110** for further processing in accord with the operating system upgrade routine.

As noted above, the HDT **230** switches DS3's and instructs the set-top **100** what VPI/VCI values to use to capture cells for particular programs. The set-top **100** processes an assigned one of the three DS3 signals carried on the coaxial cable into the VIU's premises **236**; and from that DS3, the set-top **100** captures cells having the VPI/VCI value that the HDT instructed it to capture. When a subscriber wants to switch channels, if the newly selected channel is in the DS3 stream already going to the subscriber's set-top, the HDT **230** provides the set-top with the new VPI/VCI value and PID value for the new channel. The set-top **100** can begin immediately capturing and processing cells bearing the new VPI/VCI and decoding payload data from those cells to present the program to the user via the television set or perform a software download operation, as necessary. If the selected channel is not in the DS3 currently going to the particular set-top **100**, then the HDT **230** will switch the DS3 for the new channel from the correct incoming optical fiber onto the DS3 assigned to the set-top on the fiber **233** to the ONU **232**. The ONU **232** supplies that new DS3 to the set-top via the currently assigned DS3 slot on the subscriber's coaxial drop cable **234** so that the set-top **100** will begin receiving the new DS3. Through the downstream signaling channel, the HDT **230** also informs the set-top of the new VPI/VCI to permit the set-top to capture and process cells and begin decoding payload data from those cells to present the newly selected program to the user via the television set.

The presently preferred network embodiment also provides transport for interactive broadband services such as video-on-demand and software downloading through point-to-point connections in accord with the disclosure of the 08/250,791 parent application. Typically, the network car-

ries such services offered by two or more IMTV VIPs. As shown in FIG. 8B, an ATM switch **250** provides a bit stream carrying one or more DS3's containing ATM cell streams to each HDT **230**, to provide point-to-point connections for such services. As discussed in more detail below, the access through this switch **250** is controlled by the Level 1 Gateway (L1 GTWY) **252**.

Each non-broadcast or IMTV service provider preferably has a Level 2 Gateway (L2 GTWY) **260** and some form of file server **262**. Typically, the VIP will store volumes of MPEG II encoded material in a variety of memory devices which form the server. An IMTV VIP's equipment preferably outputs ATM cell streams, encoded in the manner discussed above with regard to FIGS. 1 and 2, to the ATM switch **250** for transmission through the network. Alternatively, if the provider's equipment transmits only MPEG II bit stream data, the network operator would supply an interworking unit similar to the ATM multiplexer **29** discussed above to convert the service provider's bit stream data into a DS3 containing one or more ATM cell streams compatible with the Full Service Network. The ATM switch **250** transmits selected ATM cells on the one or more DS3 streams through an STT and an optical fiber going to the HDT **230** serving a particular VIU who requested a session with the particular IMTV VIP. As part of its routing operations, the ATM switch **250** performs policing and grooming functions of the type performed by the ATM edge device **226**.

To establish a broadband communication session or connection through the network between an interactive information service provider **210'** and a particular set-top **100** requires establishment of a virtual circuit through the ATM switch **250** and the appropriate HDT **230**. In the network of FIGS. 8A and 8B, a PVC controller **254** stores data tables defining all possible virtual circuits through the ATM switch **250** to the HDT's **230**. These data tables define the header information and the particular fiber output port used to route cells to the correct HDT **230**. The data tables thus define "permanent virtual circuits" (PVC's) between the providers and the set-top terminal devices. The PVC controller includes current VPI/VCI data available to each VIP and an ongoing record of which VPI/VCI values are in use. Thus, at any given time the PVC controller **254** knows what VPI/VCI values are available and can be assigned dynamically to provide requested bandwidth for each new IMTV session.

When a subscriber initiates a session with a broadband interactive service provider, the subscriber's set-top **100** provides an appropriate "off-hook" signal to the HDT **230**. The HDT **230** sends the message through the X.25 packet switched network **246** to the Level 1 Gateway (L1 GTWY) **252**. When the Level 1 Gateway **252** receives the addressed message from the HDT **230**, that Gateway uses the billing number ID of the set-top included in the message to check its internal database to determine if the caller is a valid network customer. If the caller is not a valid customer, the system tears downs the session. If the caller is a valid customer, the Level 1 Gateway **252** transmits a call accept message back to the set-top terminal **100** and waits for the first application level message. Once the call is accepted, an X.25 signalling link is provided between the HDT **230** and the Level 1 Gateway **252** for purposes of carrying signaling information between the set-top **100** and that Gateway, and the HDT **230** internally associates that signaling call with the signaling call over the fibers **233**, **237** and the coaxial cable **240**, i.e. the signaling link set up from the HDT **230** through the ONU **23** to the set-top **100** when the user turned on the set-top terminal.

The set-top **100** next sends an initiation or 'hello' message to the Level 1 Gateway **252** that includes basic information including the set-top ID and a set-top type designation. The Level 1 Gateway **252** interacts with the subscriber through the set-top **100** to obtain a selection identifying one of the IMTV services providers.

The Level 1 Gateway **252** may execute a PIN number access routine, if the subscriber has previously requested such access control for selected ones of the VIPs. For simplicity here, it is assumed that the currently selected VIP is not subject to such a PIN number requirement. The Level 1 Gateway **252** is merely expecting to receive the VIP selection input from the set-top **100**, e.g. in response to a menu display, within a predetermined period following the menu transmission. If the Level 1 Gateway **252** receives the selection input message from the set-top **100** within the predetermined period, the Level 1 Gateway **252** translates that message into the 4 digit code for the selected VIP's Level 2 Gateway **260**.

Once the selection of the VIP is complete, the Level 1 Gateway **252** then goes over the X.25 network **246** to communicate with the selected VIP's Level 2 Gateway **260** and indicates that it has a customer calling. The Level 1 Gateway **252** identifies the customer to the Level 2 Gateway **260** by sending the 10-digit billing number for the calling set-top **100**. The Level 1 Gateway **252** also transmits set-top identification information and set-top type information for the particular set-top **100** to the Level 2 Gateway **260**. The VIP's Level 2 Gateway **260** may accept or reject the call after receiving the initial request indicating availability of the identified customer. If accepted, the Level 2 Gateway **260** sends a message back to the Level 1 Gateway **252** indicating acceptance of the call and provides the Level 1 Gateway **252** with a server output port identification for the port on the server **262** which will service the broadband call. The Level 2 Gateway **262** may also indicate that one of a plurality of VPI/VCI values assigned to this VIP should be allocated to this particular session.

In response, the Level 1 Gateway **252** transmits the X.121 address of the calling customer's set-top **100** to the Level 2 Gateway **262**. The Level 2 Gateway **262** uses that address to initiate a new signaling communication through the X.25 network to the HDT **230** and the set-top **100**. The signaling call to the level **1** gateway **252** is taken down when Level 2 Gateway initiates its signaling call, after which the HDT **230** associates the new X.25 signaling call from the Level 2 Gateway **262** with the established signaling call between the HDT **230** and the calling subscriber's set-top **100** and performs any necessary protocol conversion. For example, the HDT places downstream signaling data from the X.25 call in packets identified with the signaling call ID assigned to the subscriber's set-top **100** at turn-on and transmits those packets through the downstream fiber **233** to the ONU **232** and the coaxial cable **240** to the subscriber's premises **236**.

The Level 1 Gateway **252** interacts with the PVC controller **250** to obtain the bandwidth through the ATM switch **250**. The Level 1 Gateway **252** advises the PVC controller **254** of the server port identification. The Level 1 Gateway **252** may advise the PVC controller **254** of the VPI/VCI value if such was assigned to the session by the Level 2 Gateway **262**. Alternatively, the Level 1 Gateway **252** may obtain a VPI/VCI for the call from internal data or from the PVC controller **254**, and then the Level 1 Gateway **252** supplies the assigned VPI/VCI to the Level 2 Gateway **262**. The Level 1 Gateway **252** also informs the HDT **230** of a DS3 and VPI/VCI value on the fiber from the ATM switch **250** to the HDT **230** which is assigned to this broadband

interactive call. If the ATM switch **250** translates the VPI/VCI values, the HDT **230** may specify the VPI/VCI value assigned to this broadband call on the DS3 on fiber to HDT **230**. Alternatively, the level 1 gateway **252** or the PVC controller **254** may administer the VPI/VCI on that fiber and provide this value to the HDT **230**.

The HDT **230** switches the identified DS3 from the fiber from the ATM switch **250** to the DS3 assigned to the subscriber's set-top on the fiber **233**. The HDT **230** also transmits a signaling message to the set-top **100** indicating the correct VPI/VCI for the cells carrying the downstream broadband transmissions. This procedure establishes a virtual circuit through the network as a logical point-to-point communication link between the selected IMTV VIP's equipment and the calling set-top **100**. The HDT **230** may provide the DET with an initial PID value for use in decoding MPEG packets for the session. Preferably, the level **2** gateway and server interact with the set-top **100** through the signaling channel and the ATM virtual circuit to download an application to the memory in the DET **102**. As part of this download, the level 2 gateway supplies either a program number of a PID value needed to decode the MPEG stream. After complete downloading of the IMTV application, the DET boots up into that application. The set-top **100** will process the ATM cells and decode MPEG data carried in those cells in a manner similar to the processing of broadcast service cells, discussed above.

During the interactive communication session between the subscriber and the IMTV VIP, the set-top **100** can transmit control signalling upstream through the ONU **232**, the HDT **230** and the X.25 data network to the level 2 gateway **260**. The level 2 gateway **260** can also send signaling information, such as control data and text/graphics, downstream through the same path to the DET **238** or preferably as user data inserted in the MPEG II broadband data stream. For downstream transmission, the server **262** and/or an associated interworking unit (not shown) will provide ATM cells with an appropriate header. The ATM switch **250** will route the cells using the header and transmit those cells to the HDT **230** serving the requesting subscriber **236**. The HDT **230** will recognize the header as currently assigned to the particular set-top **100** and will forward those cells through the downstream fiber and the ONU **232** to that set-top, in essentially the same manner as for broadcast programming.

When a session is first set up, the set-top **100** transmits an initial message identifying itself and identifying the current versions of various modules of the operating system the DET microprocessor is running. The level 2 gateway **262** examines the operating system module information and determines whether each module corresponds to the version necessary to run the particular VIP's application programs. For example, if the application programs call for version 1.1 of the graphics driver, those applications may not run properly on a DET still using version 1.0 of that module of the operating system. The identification of the operating system modules therefore permits the level 2 gateway **252** to determine if the DET **102** is running operating system modules compatible with the VIP's programs. The level 2 gateway may be able to download updated operating system software through the point-to-point connection using an appropriate bit-pattern signal corresponding to LOCK2 (FIG. 7), if this IMTVVIP is the party who originally provided the particular type of set-top terminal. If the level 2 gateway identifies one or more modules of the operating system needing to be changed to an earlier or later version thereof to achieve compatibility with the VIP's applications, the level 2 gateway will transmit an operating system rewrite code and the bit pattern through the signaling portion of the network to the microprocessor **110**. Subsequent data received over the broadband channel is routed to the microprocessor **110** and used to rewrite appropriate portions of the operating system.

For a selected VIP's level 2 gateway without access to the operating system but requiring one or more module versions not currently resident in the non-volatile memory **121** in the DET **102**, that level 2 gateway would instruct the DET to access a neutral third party level 2 gateway and server or to initiate an operating system download using a broadcast channel, to obtain the module(s) needed for compatibility with the selected VIP's service applications. The DET **102** would execute the operating system download, obtain the requisite operating system modules, and then initiate another session with the selected VIP's level 2 gateway and server.

If the operating system is compatible, the VIP's equipment **260**, **262** will download application software to the DET **102**. In the memory management illustration of FIG. 7, any L2 level 2 gateway **262** and associated server **260** can download application program software to the DRAM application memory in the DET. The downloaded applications software controls a wide variety of DET functions in accord with each VIP's services. For example, this software may specify the functionality of the user interface (UI), types of transactions to be performed, graphics styles, etc. Once all necessary software resides in memory in the DET, the user begins interaction with the services offered by the particular service provider or VIP.

The downloaded software from one service provider or VIP might present menus and prompts in simple text form. Another provider, however, might choose to present menus and prompts in a much more graphical form approaching virtual reality. One user interface for interactive services might emulate a shopping mall. The precise presentation to the user displayed on the television set is determined by the application software downloaded by the service provider and stored in the DET's system memory **120**.

The Full Service Network illustrated in FIG. **8A** and **8B** will also provide narrowband transport for voice and narrowband data services. A digital switch or an analog switch (not shown) will provide standard type plain old telephone service (POTS) for customers of the Full Service Network. The digital POTS switch provides a DS1 type digital input/output port through interfaces conforming to either **TR008** or **TR303**. The DS1 goes to the HDT **230**. The DS1 may go through a digital cross-connect switch (DCS) for routing to the various HDT's or directly to a multiplexer (not shown) serving a particular HDT **230**. The multiplexer may also receive telephone signals in DS1 format from an analog switch through a central office terminal. The central office terminal converts analog signals to digital and digital signals to analog as necessary to allow communication between the analog switch and the rest of the network.

Although not shown, the telephone service multiplexer for an HDT **230** may multiplex a number of DS1 signals for transmission over one fiber of an optical fiber pair to the HDT **230** and to demultiplex signals received over the other fiber of the fiber pair. The fiber pairs between the HDT **230** and the ONU's **232** will also have a number of DS1 channels to carry telephone and narrowband data signals to and from the subscriber's premises. In addition to the video services discussed above, the ONU **232** will provide telephone signals and appropriate power to the subscribers' premises over the twisted wire pairs **234** connected to subscribers' telephone sets **235**.

A specific example of a procedure for a broadcast upgrade of the operating system in the DET **102** of FIG. **6** through the preferred network of FIGS. **8**A and **8**B will now be discussed with regard to the flow chart of FIG. **9**.

The operating system upgrade can be initiated either automatically or manually. The DET microprocessor **110** may automatically check some time or cycle value, preferably the passage of a predetermined time interval (number of days). As another example, the DET microprocessor **110** could count the number of times the user turns off the terminal. The microprocessor will initiate the operating system upgrade routine after some specified number of power-off cycles. Alternatively, the user may execute a specified sequence of keystrokes on the remote control to call up a menu. One option on the menu is operating system upgrade. Manual selection of the operating system upgrade feature from the menu would trigger execution of the software upgrade routine.

Once initiated (step S1 in FIG. **9**), the only difference in the two procedures is whether the DET provides on screen displays during the upgrade procedure. During a manually initiated procedure, the DET will output some form of 'Please Wait' message for display on the screen of the associated television set **103**. During an upgrade procedure automatically initiated after a power-off input by the user, the output drivers of the DET **102** are off and there are no display messages unless the user turns the set-top back on before completion of the upgrade procedure.

When initiated, the DET **102** executes a normal channel selection (step S2) including the necessary interaction with the HDT to select and receive network logical channel **0**. Once 'tuned'to channel **0**, the DET microprocessor **110** calls and executes the upgrade routine from memory (step S3). The upgrade routine includes the information and instructions necessary to extract the operating system information from the MPEG data stream for the selected channel **0**. As noted above, the operating system preferably is an OS-9 based operating system. The operating system upgrade routine, however, is an assembly language program and does not require any of the operating system capabilities to execute.

To receive channel **0**, the set-top **100** interacts with the HDT **230** as discussed in detail above so that the HDT routes a DS3 containing channel **0** to the NIM **101**. The NIM **101** executes ATM cell processing to capture the cells containing the VPI/VCI value assigned to channel **0** from that DS3 and to reconstruct the MPEG II packets from the payload data of those cells. Specifically, the NIM **101** reconstructs a 6 Mbits/s MPEG II transport stream containing the video and audio information for the program guide service as well as the data for the software downloading service and supplies that stream to the DET **102**.

In a typical network operated over some substantial period of time, vendors will supply a number of different models of set-tops **100** for connection to the network. Some different types of set-tops will require different operating systems. In the preferred embodiment of the present invention, the software data carousel from server **12** contains operating system files for a plurality of different types of set-top terminal devices **100**. Each DET produced by a different manufacturer would identify and capture a different operating system file contained in the repeating carousel transmission by a PID value assigned to that manufacturer's DET. The carousel will also include information identifying the version number for each type of set-top and identifying the appropriate PID value for the packets containing that operating system. Each type of DET would recognize when the version broadcast on the network is different from that running in the DET and utilize the assigned PID value to recognize and capture the operating system for that DET.

More specifically, the MPEG II standard defines a 'network' table which may be up to 256 characters in length. In accord with the present invention, the provider of the operating system download service would populate the network table with information identifying the operating system version number and the PMT PID value assigned to each type of set-top **100**. The MPEG bit stream for channel **0** supplied to the DET **102** by the NIM **101** will contain a program association table identified by PID **0** as well as the network table. Part of the information in the program association table will identify the PID value for the network table.

After initiating execution of the upgrade routine (step S3), the DET **102** proceeds to check the version number for the particular type of set-top terminal **100** being broadcast in the carousel data stream (step S4). More specifically, the DET **102** captures the packet identified by PID **0**. From the program association table in that packet, the DET **102** identifies the PID value for the network table. Using that PID value, the DET recognizes and captures the network table packet from the data stream. The DET microprocessor **110** examines the data in the network table associated with the particular type of set-top to identify the current operating system version number being broadcast for the particular type and/or model of set-top terminal. The system memory **120** also stores a version number for the operating system the DET microprocessor **110** is currently running. The DET microprocessor **110** compares the operating system version number in the network table with the operating system version number stored in its associated system memory **120** to determine whether or not they match. If they match, an operating system upgrade is not necessary at this time, and the processing routine is complete. Therefore processing branches to step S10; and the microprocessor reboots the existing operating system from NVRAM 121, executes the normal set of associated self-diagnostics, and completes the upgrade processing.

However, if in step S4 the version numbers do not match, then the DET proceeds to extract data, etc. to upgrade the operating system with the version currently being provided by the network (step S5). The version check decision as to whether or not to complete the upgrade works both backward and forward to conform the current version running in the DET **102** to the version offered on the network. The DET will proceed with the upgrade if the version number on the network is different from the currently stored version, whether the currently stored version number is lower (older) or higher (newer) than the number of the version on the network.

To perform the extraction step, the DET **102** needs the PID value for the packets containing the relevant operating system file. The data in the program map table identifies the PID value(s) for the packets containing the download data for the operating system upgrade. The necessary PID value may be stored in memory, either through signaling communication between the set-top **100** and the HDT **230** or as part of the upgrade routine. Alternatively, the DET **102** may store a program number and access the program association table (PID **0** packet) in the stream, to obtain the PID value for the program map. In the preferred embodiment, however, the DET **102** captures this information from the network table packet. More specifically, the DET microprocessor **110** processes the network table to identify the PID value

assigned to the particular type and/or model of set-top terminal device. Using the relevant PID value from the network table packet, the MPEG demux **127** recognizes and captures the packets containing the operating system data file for the particular type of set-top **100** and forwards those packets to the DET microprocessor **110**.

In the preferred embodiment, the data is formatted into individual blocks, one of which is identified as the start of the data file. The microprocessor **110** utilizes the 'start' indication to capture one copy of the entire data file in sequence in the RAM **122**. Alternatively, the blocks could include a number identifying their location in the sequence, allowing the DET to capture blocks out of sequence, store the blocks and rearrange the blocks to compile the entire desired sequence.

The non-volatile RAM **121** comprises 1 Mbyte of memory, although sector **0** is write-protected and does not store a downloadable portion of the operating system. An MPEG II packet stream containing the operating system and associated overhead (packet headers, program map, etc.) may require 1 Mbyte of transport data. Transmission of such an operating system would require 8 Mbits of transmitted information. Assuming, as a simplified example, a 1 Mbits/s throughput rate for the downloading, the entire operating system file therefore requires 8 seconds to transmit. The transmission is continuously repeating. Assuming for a moment that only one operating system is repeating in the data carousel transmission, if the DET microprocessor attempts to capture the operating system just after the 'start' identifier, the microprocessor may have to wait almost an entire 8 seconds before receiving the 'start' identifier at the beginning of another repetition of the operating system transmission. Using a sequential acquisition beginning with the 'start' signal therefore could require as much as 16 seconds to complete, in the worst case.

The more operating systems carried on the data carousel application, the more time is required to cycle through the entire sequence of data files to capture the complete file desired. Continuing with the simplified example, if the broadcast carousel contained three operating system files of approximately the same size, the worst case time to capture the complete file might be 48 seconds.

The DET microprocessor **110** loads the captured operating system file into volatile RAM **122**. The DET microprocessor **105** then performs a checksum operation on the data file to determine if there are any errors in the received data (step **S6**). If the checksum result is not valid, indicating errors in the extracted copy of the operating system stored in the RAM **122**, then the microprocessor **110** returns to step **S5** and again extracts the relevant operating system file from the broadcast carousel. Although the separate steps are not illustrated, the microprocessor **110** will repeat steps **S5** and **S6** up to some predetermined number of times. If extraction is not successful as indicated by a valid checksum in step **S6** by the predetermined number of attempts, the microprocessor will terminate running of the upgrade routine and will reboot the existing operating system still stored in the NVRAM **121**.

The operating system file downloaded through the network also includes a bit pattern code used to indicate that the data is a valid operating system for the particular type of set-top to use to upgrade the operating system. The upgrade routine stored in ROM **115** in the DET **102** will include this bit pattern code. As part of the checksum procedure **S6** or as a separate step, the microprocessor **110** compares the bit pattern from the broadcast operating system now loaded in RAM **122** to the valid bit pattern stored in ROM **115**. If the received code matches the stored code, then the microprocessor will proceed with the upgrade operation. If there is no match, the microprocessor may repeat the extraction procedure one or more times or terminate the upgrade routine.

If the checksum result in step **S6** is valid (and the associated bit pattern check is valid), there are no errors in the extracted operating system, and the microprocessor **110** halts execution of the old operating system running from NVRAM **121** (step **S7**). Assuming that the non-volatile memory **121** is a flash memory, the running of the existing operating system is terminated by shutting down operation of the memory. Under control of the assembly language upgrade routine, the microprocessor **110** copies the operating system, sector by sector, from volatile RAM **122** into the flash memory non-volatile RAM **121** (step **S8**). In the currently preferred implementation, it takes approximately 30 seconds to write 1 Mbyte of flash memory.

As noted above with regard to FIG. **7**, a specified bit pattern is needed to authorize overwriting of certain sectors of the flash memory **121**. The operating system upgrade routine, preferably stored in ROM **115**, includes the bit pattern needed to authorize overwriting of all sectors except sector **0**. In operation, each sector of flash memory that is to be rewritten is erased, and new code is written into that sector from the RAM **122**.

Once the operating system is fully loaded into the sectors of the flash memory, then the microprocessor executes another checksum operation (step **S9**). If the checksum operation produces a 'valid' result indicating no errors are present in the operating system now loaded into flash memory **121**, the operating system has been successfully loaded, and the microprocessor **110** therefore initiates a reboot routine (step **S10**). As a result of the reboot, the microprocessor begins running the new operating system from the flash memory, and the upgrade procedure is complete.

In the disclosed operation, the microprocessor **110** attempts to write the operating system from the volatile RAM **122** to the non-volatile RAM **121** (flash memory) only if the checksum procedure executed after extraction of data from the broadcast (step **S6**) indicates reception and extraction of an error free copy of the operating system. The copy of the operating system in the RAM **122** therefore should be error free, and a failure to correctly write the operating system to the flash memory generally results from a temporary writing error. If the checksum following a write operation to the flash memory fails, then the microprocessor **110** returns to step **S8** and attempts to reload the operating system into the flash memory **121** from the volatile RAM **122**.

If the upgrade procedure began automatically after the user turned the set-top **100** 'OFF', then the DET **102** returns to its low power stand-by state after reboot. If the upgrade procedure began as a result of the manual activation, then the DET **102** utilizes the last channel viewed data to resume audio/video output. Since the upgrade routine utilized a channel selection procedure for channel **0**, the DET **102** will begin outputting the signals to display the audio/video information for the program guide from that channel.

As noted above, in a manually activated procedure, the DET will produce a 'Please Wait' type display on the associated television set **100**. Although not shown as a step in the process of FIG. **9**, this display to the user is initiated prior to bringing down the operating system. Specifically, while still running under the old operating system, the

microprocessor **110** provides data through the graphics overlay controller **133** to fill the video RAM **135** with the desired display information. While the operating system is down and being rewritten, the contents of the video RAM **135** would remain unchanged and cause the display to persist on the output to the associated television **103**.

When the reboot operation is complete, the DET begins execution of the new operating system. As a result, the DET will replace the 'wait' display with new video information from the last selected channel. In the present circumstances, the last channel selected through an interaction with the HDT was channel **0**. However, since the DET is now running the operating system instead of the software upgrade program, the DET will search for and decode packets containing audio and video information, in the normal manner, and ignore packets having PID values associated with data. After the reboot, the DET therefore will begin displaying video from that same channel, albeit by processing the video and audio packets from the MPEG stream. As noted above, the actual video and audio on this channel preferably relate to a program guide service.

As noted above, when the user turns the box 'OFF', the microprocessor **110** turns off the various output drivers thereby terminating the television display functionality. To the user, the set-top **100** appears off. The microprocessor **110**, however, remains powered and as part of its self-diagnostic routine can determine whether or not an operating system upgrade is necessary. If an upgrade is automatically initiated, e.g. after 10 'OFF' cycles by user command, the set-top **100** can interact with the HDT **230**, check the network operating system version number for the particular type of set-top and execute the operating system upgrade procedure as discussed above, without the user being aware. If the user turns the set-top **100** 'ON' during the automatic upgrade procedure, the DET **102** will provide the please wait display, complete the upgrade procedure and boot up in the new operating system.

While this invention has been described in connection with what is presently considered to be the preferred embodiments, it is to be understood that the invention is not limited to the disclosed embodiments, but, on the contrary, is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims.

We claim:

1. A set-top terminal device comprising:

a network interface module adapted to couple the terminal to a communication network for receiving at least selected ones of a plurality of broadcast digital broadband channels at least one of which carries audio/video program information in compressed, digital form in packets of a standardized format and at least one of which carries cyclically repetitive transmissions of operating system software in packets of the standardized format, wherein said network interface module receives an Asynchronous Transfer Mode (ATM) cell stream and extracts packets of the standardized format from payloads of ATM cells; and

a digital entertainment terminal comprising:

(a) an audio/video processor responsive to at least some of the packets extracted by the network interface module for processing the compressed, digital audio/video program information;

(b) a memory;

(c) means for receiving inputs from a user; and

(d) a control processor controlling operations of the set-top terminal;

wherein said control processor captures said operating system software from at least some of the packets extracted by the network interface module for one of the selected digital broadband channels within a transmission cycle, loads the captured operating system software into the memory and begins operation in accord with the operating system software loaded into the memory, said control processor controlling the network interface module and the audio/video processor in response to the user inputs in accord with the operating system software loaded in said memory.

2. A device as recited in claim **1**, wherein said memory comprises a non-volatile random access memory.

3. A device as recited in claim **2**, wherein said non-volatile random access memory comprises a flash memory.

4. A device as recited in claim **1**, wherein said digital entertainment terminal further comprises a random access memory storing applications software for use by said control processor while running the operating system software.

5. A device as recited in claim **1**, wherein said audio/video processor comprises:

an audio decoder for decoding compressed, digital audio information;

a video decoder for decoding compressed, digital video information; and

a packet demultiplexer for analyzing packet identifiers contained in the packets of the standardized format to identify packets containing compressed, digital audio information and to route information from those packets to the audio decoder, to identify packets containing compressed, digital video information and to route information from those packets to the video decoder, and to identify packets containing operating system software and route software from those packets to the control processor.

6. A device as recited in claim **5**, wherein:

the audio decoder comprises an MPEG audio decoder;

the video decoder comprises an MPEG video decoder; and

the packet demultiplexer is an MPEG demultiplexer.

7. A device as recited in claim **5**, wherein said network interface module supplies the extracted packets to the packet demultiplexer.

8. A device as recited in claim **1**, wherein said digital entertainment terminal further comprises a memory storing a routine which the control processor executes to control capturing of the operating system software.

9. A communication system comprising:

a source system comprising:

(a) a program source supplying a broadband program signal,

(b) a software server cyclically outputting a data file containing an operating system, and

(c) an encoder system for packetizing the broadband program signal and the data file in digital packets of a standard format, wherein said encoder system comprises an encoder for digitizing and compressing the broadband program signal into program data and encapsulating the program data in a sequence of packets of the standard format, a data module for encapsulating the data file containing an operating system in a sequence of packets of the standard format, and an Asynchronous Transfer Mode (ATM) multiplexer for combining the packets containing the broadband program information and the packets containing the data file into a single stream for broadcast through the network on a single one of the channels;

a digital network broadcasting a plurality of digital broadband channels, said digital network receiving and broadcasting the digital packets from the encoder system on at least one of the channels; and

a plurality of set-top terminal devices, each set-top terminal device comprising:

(1) an interface coupled to the digital network for receiving at least a selected one of the channels, selectively including at least one channel carrying packets containing the broadband program information, and at least one channel carrying packets containing the operating system data file;

(2) a program signal processor for processing the packets containing the broadband program information;

(3) a memory;

(4) means for receiving inputs from a user; and

(5) a control processor controlling operations of the set-top terminal;

wherein said control processor captures said operating system data file from a selected one of the digital broadband channels, loads the captured operating system into the memory and begins operation in accord with the operating system loaded into the memory, said control processor controlling the interface and the program signal processor in response to the user inputs in accord with the operating system loaded in said memory.

10. A communication system as recited in claim **9**, wherein said encoder for digitizing and compressing the broadband program signal comprises a real time encoder for digitizing and compressing an audio/video program signal.

11. A communication system as recited in claim **10**, wherein said real time encoder comprises an MPEG encoder.

12. A communication system as recited in claim **9**, wherein said digital network comprises:

a first optical fiber receiving the digital packets from the encoder system;

a second optical fiber receiving packets containing broadband program information from another source system;

a system of optical fibers for broadcasting the digital packets from the encoder system on at least a first one of the channels and for broadcasting the packets containing the broadband program information from another source on at least a second one of the channels; and

a plurality of host digital terminals each coupled between the system of optical fibers and a group of the set-top terminals for routing selected ones of the channels to set-top terminals in each group.

13. In a digital network broadcasting packetized audio/video program information through a plurality of digital broadband channels to a plurality of digital terminals connected to the network, a method comprising the steps of:

cyclically broadcasting an operating system together with predetermined identification data relating to the operating system on one digital broadband channel;

selectively receiving the one digital broadband channel and capturing the predetermined identification data;

comparing the captured predetermined identification data to identification data stored in one of the digital terminals;

based on the results of the comparison, capturing a copy of the operating system from the cyclical broadcast;

initiating operating of the one digital terminal in accord with the captured copy of the operating system;

receiving and storing application software in the one digital terminal via a digital communication link through the network; and

executing the application software under control of the captured copy of the operating system.

14. A method as recited in claim **13**, wherein the predetermined identification data identifies a terminal type, and the operating system is stored if the terminal type identified by the predetermined identification data matches terminal type identification data stored in the one terminal.

15. A method as recited in claim **13**, wherein the predetermined identification data identifies a version number of the operating system being broadcast, and the operating system is stored if the version number identified by the predetermined identification data is different from a version number of an operating system previously stored in the one terminal.

16. A method as recited in claim **13**, wherein the operating system comprises:

a microprocessor operating system;

at least one driver routine used by a microprocessor to control components of a terminal; and

a resident application controlling at least selection of channels through the network in response to user inputs.

17. A method as recited in claim **13**, wherein the digital communication link comprises one of the digital broadband channels.

18. A method as recited in claim **13**, wherein the digital communication link comprises a broadband point-to-point link.

19. A method comprising:

encoding a plurality of broadband program information signals as digitized, compressed data in packet streams of a standard format;

cyclically generating a first data file containing an operating system comprising code executable by a first type of terminal and a data file containing an operating system comprising code executable by a second type of terminal different in type from the first type of terminal;

forming a sequence of packets in the standard format including: packets containing the first data file and a first identifier, packets containing the second data file and a second identifier, and at least one packet containing data associating the first and second identifiers with the first and second types of terminal, respectively;

broadcasting the packet streams and the sequence of packets on a plurality of multiplexed channels;

in a receiving terminal of a predetermined type:

(a) selectively receiving a channel carrying the sequence of packets;

(b) capturing said at least one packet;

(c) identifying the first type of terminal or the second type of terminal as corresponding to the predetermined type of the receiving terminal;

(d) recognizing the first or second identifier as associated with the identified terminal type;

(e) using the recognized identifier to capture a copy of the operating system for the identified terminal type from the sequence of packets; and

(f) executing at least a portion of the code from the captured copy of the operating system for the identified terminal type to initiate operation of the receiving terminal, the operation of the receiving terminal including reception of a user selected channel car-

rying a packet stream and processing digitized, compressed data from that packet stream to present broadband program information to a user in humanly perceptible form.

20. A method as recited in claim **19**, wherein:

the step of using the recognized identifier to capture a copy of the operating system comprises storing payload data from packets containing the recognized identifier in random access memory; and

the step of initiating operation comprises transferring the payload data from the random access memory to a non-volatile memory, and booting up the receiving terminal from the payload data in the nonvolatile memory.

21. A method as recited in claim **19**, further comprising encoding another broadband program information signal as digitized, compressed data in another packet stream of a standard format,

wherein the step of broadcasting comprises:

multiplexing the sequence of packets and said another packet stream into one channel stream, and

broadcasting the one channel stream through one of the multiplexed channels.

22. A method comprising:

encoding a plurality of broadband program information signals as digitized, compressed data in packet streams of a standard format;

cyclically generating a first data file containing an operating system for a first type of terminal and a data file containing an operating system for a second type of terminal;

forming a sequence of packets in the standard format including: packets containing the first data file and a first identifier, packets containing the second data file and a second identifier, and at least one packet containing data associating the first and second identifiers with the first and second types of terminal, respectively;

broadcasting the packet streams and the sequence of packets on a plurality of multiplexed channels,

wherein the multiplexed channels comprise Asynchronous Transfer Mode (ATM) virtual circuits, each virtual circuit being identified by a different virtual path identifier/virtual circuit identifier (VPI/VCI) value;

in a receiving terminal of a predetermined type:

(a) selectively receiving a channel carrying the sequence of packets;

(b) capturing said at least one packet;

(c) identifying the first type of terminal or the second type of terminal as corresponding to the predetermined type of receiving terminal;

(d) recognizing the first or second identifier as associated with the identified terminal type;

(e) using the recognized identifier to capture a copy of the operating system for the identified terminal type from the sequence of packets; and

(f) initiating operation of the receiving terminal in accord with the captured copy of the operating system for the identified terminal type, operation of the receiving terminal including reception of a user selected channel carrying a packet stream and processing digitized, compressed data from that packet stream to present broadband program information to a user in humanly perceptible form.

23. A method as recited in claim **22**, wherein the step of selectively receiving a channel comprises receiving and processing ATM cells containing a VPI/VCI value assigned to the selectively received channel.

24. A method comprising:

encoding a plurality of broadband program information signals as digitized, compressed data in packet streams of a standard format;

cyclically generating a first data file containing an operating system comprising code executable by a first type of terminal and a data file containing an operating system comprising code executable by a second type of terminal different in type from the first type of terminal;

forming a sequence of packets in the standard format including: packets containing the first data file and a first identifier, packets containing the second data file and a second identifier, and at least one packet containing data associating the first and second identifiers with the first and second type of terminal and first and second operating system version numbers, respectively;

broadcasting the packet streams and the sequence of packets on a plurality of multiplexed channels;

in a receiving terminal of a predetermined type:

selectively receiving a channel carrying the sequence of packets;

capturing said at least one packet;

from the data in said at least one packet, identifying the first type of terminal or the second type of terminal as corresponding to the predetermined type of the receiving terminal;

from the data in said at least one packet, identifying the version number for the identified terminal type;

if the identified version number differs from a version number of an operating system previously stored in the receiving terminal, recognizing the first or second identifier as associated with the identified terminal type and using the recognized identifier to capture a copy of the operating system for the identified terminal type from the sequence of packets; and

executing at least a portion of the code from the captured copy of the operating system for the identified terminal type to initiate operation of the receiving terminal, the operation of the receiving terminal including reception of a user selected channel carrying a packet stream and processing digitized, compressed data from that packet stream to present broadband program information to a user in humanly perceptible form.

25. A method as recited in claim **24**, wherein:

the step of using the recognized identifier to capture a copy of the operating system comprises storing payload data from packets containing the recognized identifier in random access memory; and

the step of initiating operation comprises transferring the payload data from the random access memory to a non-volatile memory, and booting up the receiving terminal from the payload data in the non-volatile memory.

26. A method as recited in claim **24**, further comprising encoding another broadband program information signal as digitized, compressed data in another packet stream of a standard format,

wherein the step of broadcasting comprises:

multiplexing the sequence of packets and said another packet stream into one channel stream, and

broadcasting the one channel stream through one of the multiplexed channels.

27. A method comprising:

encoding a plurality of broadband program information signals as digitized, compressed data in packet streams of a standard format;

cyclically generating a first data file containing an operating system for a first type of terminal and a data file containing an operating system for a second type of terminal;

forming a sequence of packets in the standard format including: packets containing the first data file and a first identifier, packets containing the second data file and a second identifier, and at least one packet containing data associating the first and second identifiers with the first and second type of terminal and first and second operating system version numbers, respectively;

broadcasting the packet streams and the sequence of packets on a plurality of multiplexed channels,

wherein the multiplexed channels comprise Asynchronous Transfer Mode (ATM) virtual circuits, each virtual circuit being identified by a different virtual path identifier/virtual circuit identifier (VPI/VCI) value;

in a receiving terminal of a predetermined type:

selectively receiving a channel carrying the sequence of packets;

capturing said at least one packet;

from the data in said at least one packet, identifying the first type of terminal or the second type of terminal as corresponding to the predetermined type of receiving terminal;

from the data in said at least one packet, identifying the version number for the identified terminal type;

if the identified version number differs from a version number of an operating system previously stored in the receiving terminal, recognizing the first or second identifier as associated with the identified terminal type and using the recognized identifier to capture a copy of the operating system for the identified terminal type from the sequence of packets; and

initiating operation of the receiving terminal in accord with the captured copy of the operating system for the identified terminal type, operation of the receiving terminal including reception of a user selected channel carrying a packet stream and processing digitized, compressed data from that packet stream to present broadband program information to a user in humanly perceptible form.

28. A method as recited in claim 27, wherein the step of selectively receiving a channel comprises receiving and processing ATM cells containing a VPI/VCI value assigned to the selectively received channel.

29. A method comprising:

selectively receiving in a terminal an Asynchronous Transfer Mode (ATM) digital broadcast channel identified by a virtual path identifier/virtual circuit identifier (VPI/VCI) value and carrying a digital transport stream of packets;

capturing at least one packet of data from the digital transport stream;

from the data in said at least one packet, identifying a version number for an operating system carried in the digital transport stream;

if the identified version number differs from a version number of an operating system previously stored in the terminal, capturing the operating system from the transport stream; and

initiating operation of the terminal in accord with the captured copy of the operating system, operation of the terminal including reception of a user selected ATM channel and processing digitized, compressed data from the user selected ATM channel to present broadband program information to a user in humanly perceptible form.

30. A method as recited in claim 29 further comprising the step of initiating the method in response to a predetermined user input.

31. A method as recited in claim 29, further comprising the step of automatically initiating the method in response to a predetermined event.

32. A method as received, in claim 31, wherein the predetermined event is passage of a specified time period.

33. A method as received in claim 31, wherein the predetermined event comprises turn-off of the terminal.

34. A method as recited in claim 29, further comprising the steps of:

counting each occurrence of an 'off' instruction input to the terminal from a user; and

when the count reaches a predetermined value, initiating the method.

35. A communication system comprising:

a source system supplying a broadband program signal, and a cyclically repeating data file containing an operating system, said broadband program signal and the data file being encoded in digital packets of a standard format;

an Asynchronous Transfer Mode (ATM) digital network broadcasting a plurality of digital broadband channels in virtual circuits, each virtual circuit being identified by a different virtual path identifier/virtual circuit identifier (VPI/VCI) value, said digital network receiving and broadcasting the digital packets from the source system on at least one of the channels; and

a plurality of set-top terminal devices, each set-top terminal device comprising:

(1) an interface coupled to the digital network for receiving at least a selected one of the channels, selectively including at least one channel carrying packets containing the broadband program information, and at least one channel carrying packets containing the operating system data file;

(2) a program signal processor for processing the packets containing the broadband program information;

(3) a memory;

(4) means for receiving inputs from a user; and

(5) a control processor controlling operations of the set-top terminal;

wherein said control processor captures said operating system data file from a selected one of the digital broadband channels, loads the captured operating system into the memory and begins operation in accord with the operating system loaded into the memory, said control processor controlling the interface and the program signal processor in response to the user inputs in accord with the operating system loaded in said memory.

36. A communication system as recited in claim 35, wherein said source system comprises:

an encoder for digitizing and compressing the broadband program signal into program data and encapsulating the program data in a sequence of packets of the standard format; and

a data module for encapsulating the data file containing an operating system in a sequence of packets of the standard format.

37. A communication system as recited in claim 36, wherein said source system further comprises a multiplexer for combining the packets containing the broadband program information and the packets containing the data file into a single stream for broadcast through the network on a single one of the channels.

38. A set-top terminal device comprising:

a network interface module adapted to couple the terminal to a communication network for receiving at least selected ones of a plurality of broadcast digital broadband channels at least one of which carries audio/video program information in compressed, digital form in packets of a standardized format and at least one of which carries cyclically repetitive transmissions of operating system software in packets of the standardized format; and

a digital entertainment terminal comprising:

(a) an audio/video processor for processing the compressed, digital audio/video program information;

(b) an operating system memory;

(c) a random access memory;

(d) means for receiving inputs from a user; and

(e) a control processor controlling operations of the set-top terminal, wherein

said control processor captures said operating system software from one of the selected digital broadband channels within a transmission cycle, loads the captured operating system software into the operating system memory and begins operation in accord with the operating system software loaded into the operating system memory,

said control processor captures application software received through the network interface module, stores captured application software in the random access memory and executes the stored application software under control of the captured copy of the operating system, and

said control processor controls the network interface module and the audio/video processor in accord with the operating system software loaded in said operating system memory, and controls at least some responses to the user inputs with the application software.

39. A device as recited in claim 38 wherein said audio/video processor comprises:

an audio decoder for decoding compressed, digital audio information;

a video decoder for decoding compressed, digital video information; and

a packet demultiplexer for analyzing packet identifiers contained in the packets of the standardized format to identify packets containing compressed, digital audio information and to route information from those packets to the audio decoder, to identify packets containing compressed, digital video information and to route information from those packets to the video decoder, and to identify packets containing operating system software and application software and route software from those packets to the control processor.

40. A device as recited in claim 39, wherein:

the audio decoder comprises an MPEG audio decoder;

the video decoder comprises an MPEG video decoder; and

the packet demultiplexer is an MPEG demultiplexer.

41. A device as recited in claim 39, wherein said network interface module receives an Asynchronous Transfer Mode (ATM) cell stream, extracts packets of the standardized format from payloads of ATM cells and supplies the extracted packets to the packet demultiplexer.

42. A communication system comprising:

a source'system comprising:

(a) a program source supplying a broadband program signal,

(b) a software server cyclically outputting a data file containing an operating system, and

(c) an encoder system for packetizing the broadband program signal and the data file in digital packets of a standard format;

a digital network broadcasting a plurality of digital broadband channels, said digital network receiving and broadcasting the digital packets from the encoder system on at least one of the channels and transporting an application program through at least one digital broadband channel; and

a plurality of set-top terminal devices, each set-top terminal device comprising:

(1) an interface coupled to the digital network for receiving at least a selected one of the channels,

(2) a program signal processor for processing packets containing the broadband program information received through the interface,

(3) an operating system memory,

(4) a random access memory for storing the application program when received through the interface,

(5) means for receiving inputs from a user, and

(5) a control processor controlling operations of the set-top terminal,

wherein said control processor captures said operating system data file from a selected one of the digital broadband channels, loads the captured operating system into the memory and begins operation in accord with the operating system loaded into the memory, said control processor executing the application program from the random access memory and controlling the interface and the program signal processor in accord with the operating system loaded in said memory and controlling at least some responses to user inputs in accord with the application program.

43. A communication system as recited in claim 42, wherein said encoder system further comprises a multiplexer for combining the packets containing the broadband program information and the packets containing the data file into a single stream for broadcast through the network on a single one of the channels.

44. A communication system as recited in claim 43, wherein said multiplexer is an Asynchronous Transfer Mode (ATM) multiplexer.

45. A communication system as recited in claim 44, wherein said digital network comprises:

a first optical fiber receiving the digital packets from the encoder system;

a second optical fiber receiving packets containing broadband program information from another source system;

a system of optical fibers for broadcasting the digital packets from the encoder system on at least a first one of the channels and for broadcasting the packets containing the broadband program information from another source on at least a second one of the channels; and

a plurality of host digital terminals each coupled between the system of optical fibers and a group of the set-top terminals for routing selected ones of the channels to set-top terminals in each group.

* * * * *

US005894516A

United States Patent [19]

Brandenburg

[11] **Patent Number: 5,894,516**

[45] **Date of Patent: *Apr. 13, 1999**

[54] **BROADCAST SOFTWARE DISTRIBUTION**

[75] Inventor: **Barbara B. Brandenburg**, Columbia, S.C.

[73] Assignee: **NCR Corporation**, Dayton, Ohio

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **08/679,638**

[22] Filed: **Jul. 10, 1996**

[51] **Int. Cl.**[6] **H04L 9/00**

[52] **U.S. Cl.** **380/4**; 380/9; 380/21; 380/23; 380/25; 380/49; 380/50; 348/6; 455/3.1; 455/3.2

[58] **Field of Search** 380/4, 21, 9, 10, 380/20, 49, 50, 59, 23, 25; 705/26, 27; 348/6, 7, 8, 9, 12; 455/3.1, 3.2, 4.1, 4.2, 5.1, 6.1, 6.3

[56] **References Cited**

U.S. PATENT DOCUMENTS

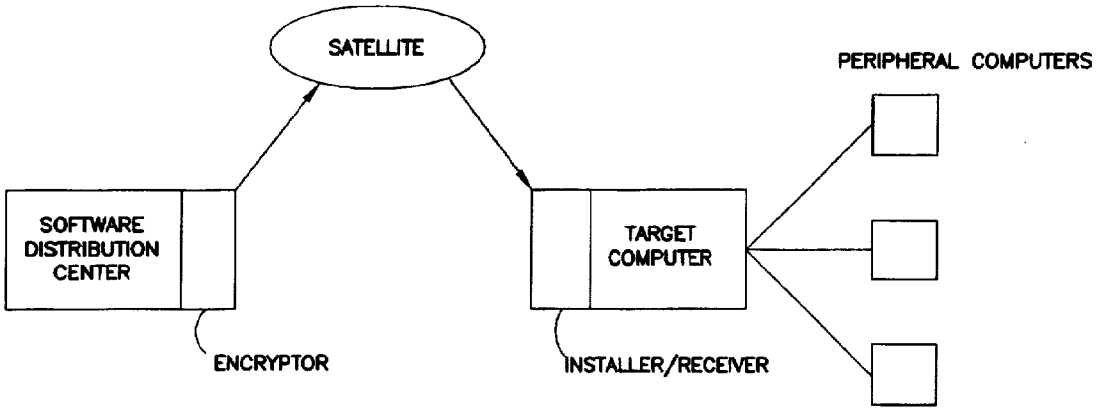
5,023,907	6/1991	Johnson et al.	380/4
5,034,980	7/1991	Kubota	380/4
5,224,166	6/1993	Hartman, Jr.	380/50
5,388,158	2/1995	Berson	380/23
5,392,351	2/1995	Hasebe et al.	380/4
5,400,403	3/1995	Fahn et al.	380/21
5,416,840	5/1995	Cane et al.	380/4
5,586,186	12/1996	Yuval et al.	380/30
5,598,470	1/1997	Cooper et al.	380/4
5,694,471	12/1997	Chen et al.	380/25
5,754,957	5/1998	Khan	455/436
5,761,301	6/1998	Oshima et al.	380/4

Primary Examiner—Bernarr E. Gregory
Attorney, Agent, or Firm—Gates & Cooper

[57] **ABSTRACT**

A method, apparatus, and article of manufacture for broadcasting encrypted software to a target computer enables simultaneous transmission to a plurality of licensed target computers. An encryption key is generated to encrypt a software package. The encryption key is then itself encrypted using a target computer identification code, and the encrypted encryption key is loaded onto the target computer. The encrypted software is broadcast, for example, via satellite, and received at the target computer. The target computer uses its identification code to decrypt the encrypted encryption key (i.e., unlock the encryption key). Once the target computer unlocks the encryption key, it uses the encryption key to decrypt the software to be installed on the target computer.

21 Claims, 4 Drawing Sheets



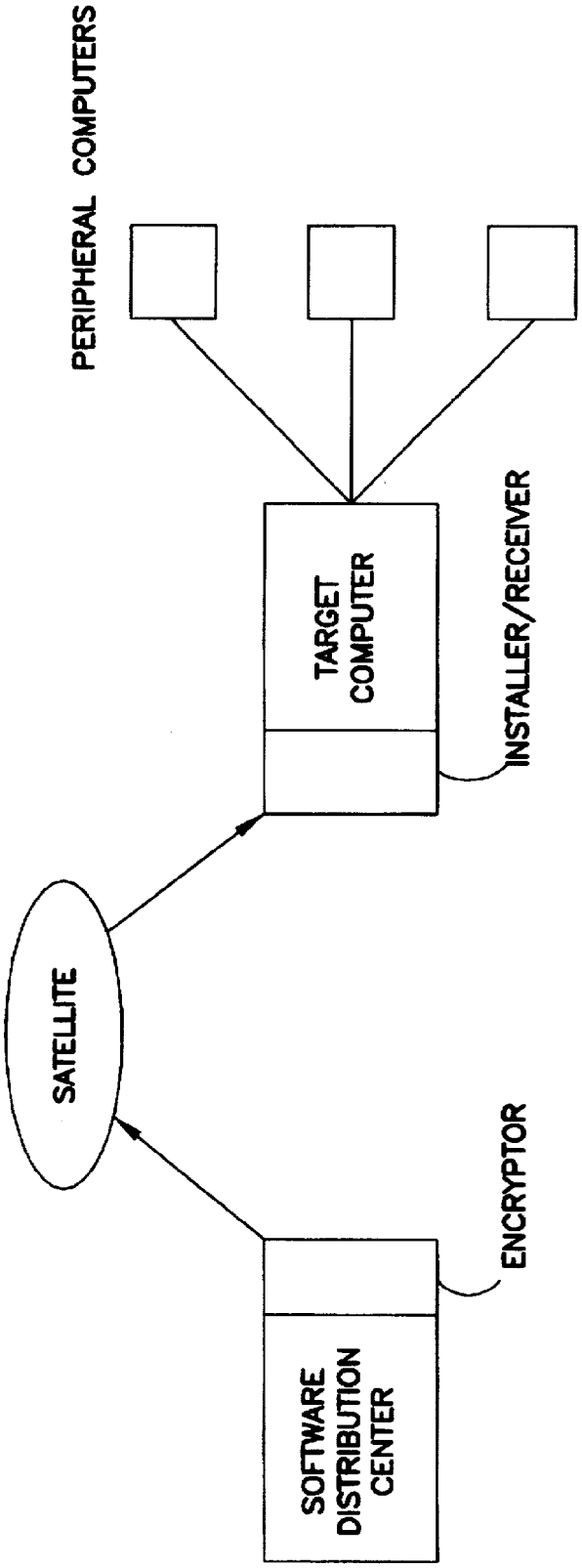

PERIPHERAL COMPUTERS
TARGET COMPUTER
INSTALLER/RECEIVER
SATELLITE
ENCRYPTOR
SOFTWARE DISTRIBUTION CENTER

FIG. 1

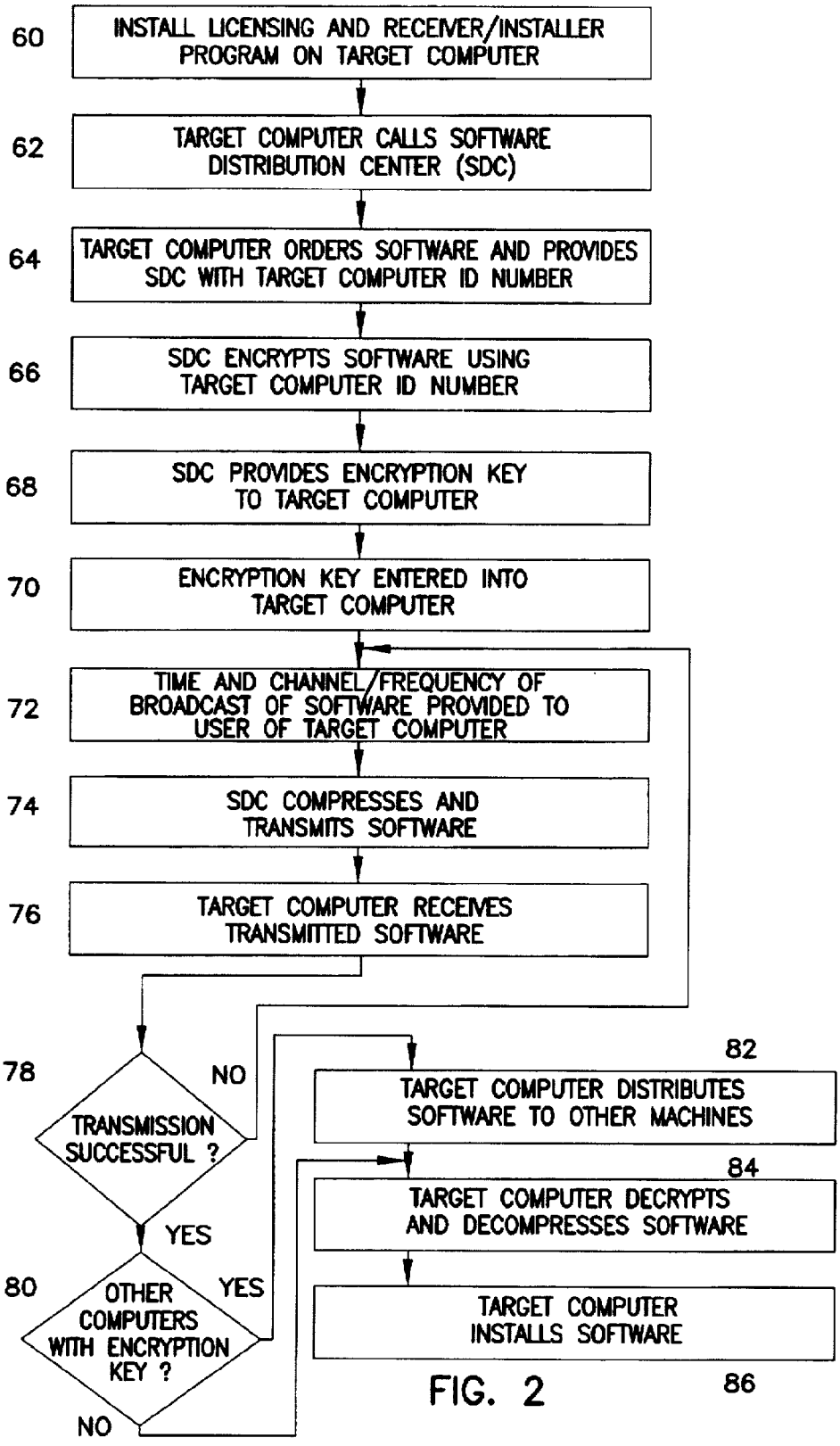

60
INSTALL LICENSING AND RECEIVER/INSTALLER PROGRAM ON TARGET COMPUTER
62
TARGET COMPUTER CALLS SOFTWARE DISTRIBUTION CENTER (SDC)
64
TARGET COMPUTER ORDERS SOFTWARE AND PROVIDES SDC WITH TARGET COMPUTER ID NUMBER
66
SDC ENCRYPTS SOFTWARE USING TARGET COMPUTER ID NUMBER
68
SDC PROVIDES ENCRYPTION KEY TO TARGET COMPUTER
70
ENCRYPTION KEY ENTERED INTO TARGET COMPUTER
72
TIME AND CHANNEL/FREQUENCY OF BROADCAST OF SOFTWARE PROVIDED TO USER OF TARGET COMPUTER
74
SDC COMPRESSES AND TRANSMITS SOFTWARE
76
TARGET COMPUTER RECEIVES TRANSMITTED SOFTWARE
78
TRANSMISSION SUCCESSFUL ?
NO
YES
80
OTHER COMPUTERS WITH ENCRYPTION KEY ?
YES
NO
82
TARGET COMPUTER DISTRIBUTES SOFTWARE TO OTHER MACHINES
84
TARGET COMPUTER DECRYPTS AND DECOMPRESSES SOFTWARE
TARGET COMPUTER INSTALLS SOFTWARE
86

FIG. 2

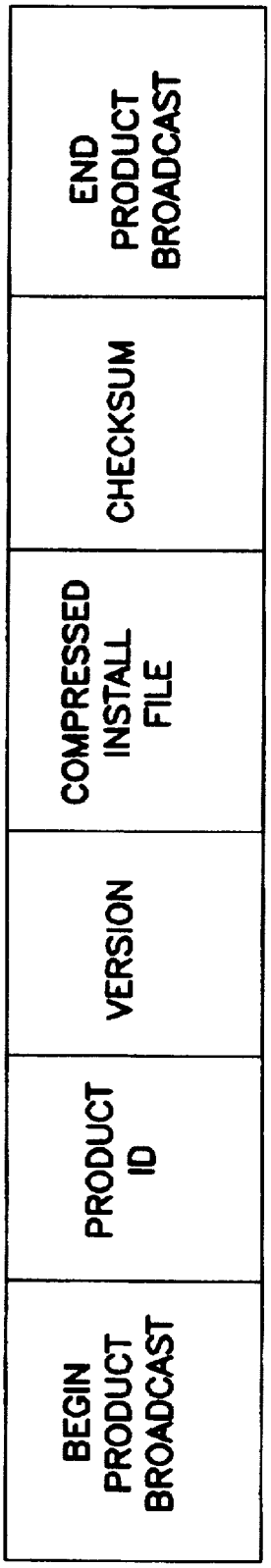
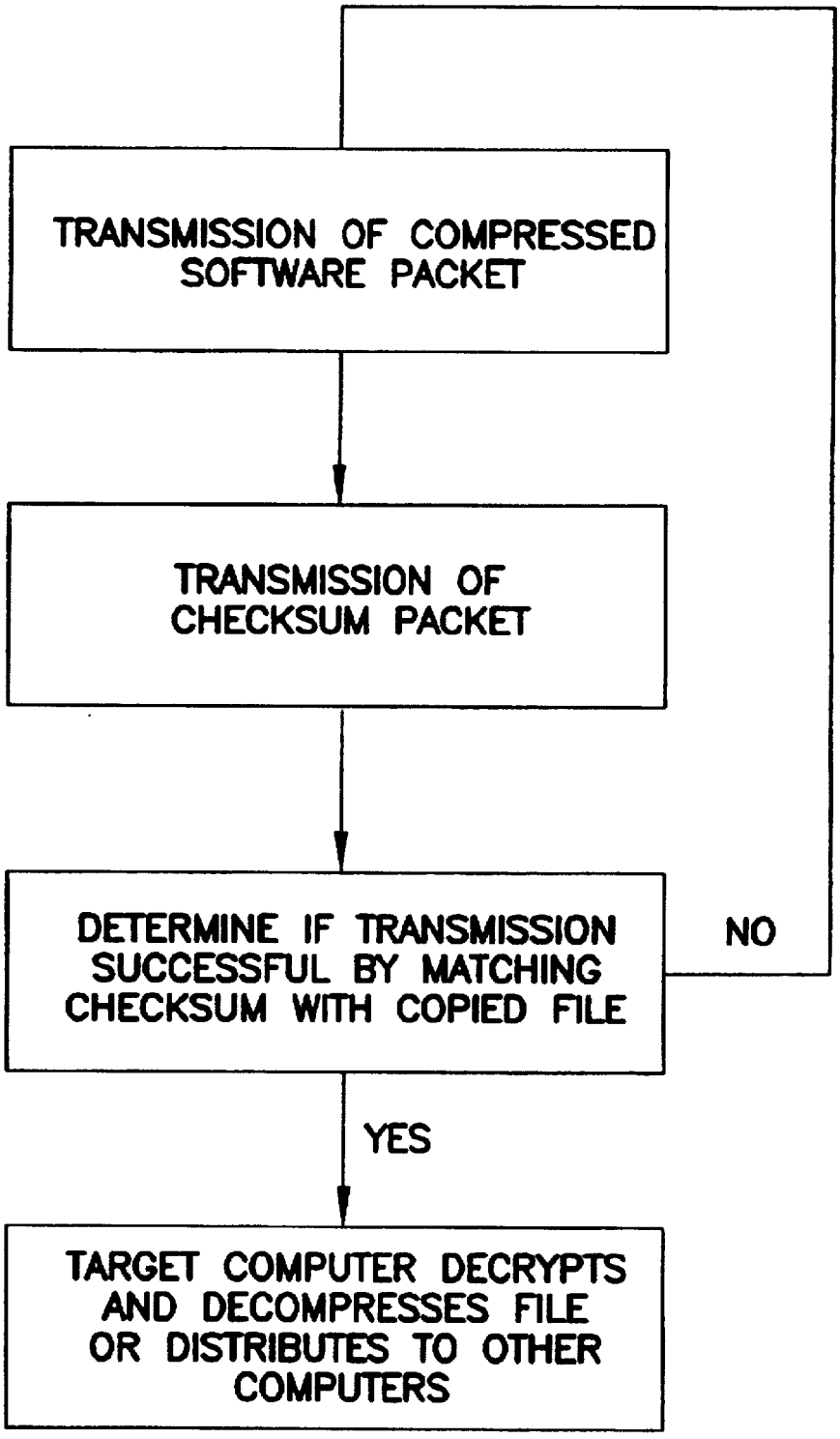

BEGIN PRODUCT BROADCAST
PRODUCT ID
VERSION
COMPRESSED INSTALL FILE
CHECKSUM
END PRODUCT BROADCAST

FIG. 3

FIG. 4


TRANSMISSION OF COMPRESSED SOFTWARE PACKET
TRANSMISSION OF CHECKSUM PACKET
DETERMINE IF TRANSMISSION SUCCESSFUL BY MATCHING CHECKSUM WITH COPIED FILE
NO
YES
TARGET COMPUTER DECRYPTS AND DECOMPRESSES FILE OR DISTRIBUTES TO OTHER COMPUTERS

BROADCAST SOFTWARE DISTRIBUTION

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates in general to broadcast distribution, and in particular, to broadcast software distribution using encryption key locking and unlocking procedures.

2. Description of Related Art

The software industry as a whole has experienced tremendous growth in recent years. There is a continuous demand for new software products that address the needs in new or changing industries. Moreover, software companies routinely upgrade previously-released software products in response to specific user needs and/or to provide a product in a more efficient manner.

Software companies have traditionally distributed its software products through physical media such as tapes or diskettes, and, more recently, CD ROM. Companies store their software on these physical media and ship them to customers for installation onto their home computers.

In distributing software by this method, however, various problems have been encountered. The cost of media duplication, shipping, and storage is quite high in many such applications. Moreover, the elimination of many types of such physical media has created unwanted environmental-waste concerns. In addition, this type of distribution involves unwanted delay associated with waiting for media copies, packing, addressing, and shipping to obtain new products or new versions of existing products.

An alternative method of distributing software is through phone lines. Software distribution centers, having a host computer with a modem, transmit the software through phone lines to a customer's computer in response to the customer's order for particular software packages.

This alternative form of distribution has also encountered problems. The number of customers who can receive a given transmission is limited by the physical capabilities of the phone lines. If numerous customers order the same software package, such as when a company distributes an upgraded version to existing subscribers, the software company must repeatedly transmit the same software package until all its customers have received the product. This approach is both costly and time consuming.

Accordingly, there is a need for an improved method and computer system for distributing software that overcomes the above-mentioned deficiencies associated with prior techniques. The present invention provides a solution to these and other problems, offering advantages over conventional implementations.

SUMMARY OF THE INVENTION

To overcome the limitations in the prior art described above, and to overcome other limitations that will become apparent upon reading and understanding the present specification, the present invention provides a unique method, apparatus, and article of manufacture for broadcasting encrypted software to a target computer. In one embodiment of the present invention, an encryption key, which is unique to the particular software package, is used to encrypt the software package, and is then itself encrypted using the unique identification code of the target computer. This key is loaded onto the target computer, to lock the particular software package to the target computer. The software is then broadcast from a send computer and received at the target computer. In a more particular implementation, the software is broadcast via satellite. The target computer obtains the encryption key by decrypting the encryption key using its identification code. The encryption key is then used to decrypt the software to enable the software to be installed onto the target computer.

BRIEF DESCRIPTION OF THE DRAWINGS

Referring now to the drawings in which like reference numbers represent corresponding parts throughout:

FIG. **1** illustrates an exemplary computer system environment for use in accordance with the present invention;

FIG. **2** is a flow chart illustrating exemplary steps which may be used to program the computer system of FIG. **1**, according to the present invention;

FIG. **3** illustrates the packets of a compressed file that are transmitted to a target computer in a broadcast session, according to the present invention; and

FIG. **4** is a flow chart illustrating exemplary steps, according to the present invention, for verifying that software has been successfully transmitted in the system of FIG. **1**.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

In the following detailed description of the drawings, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration a specific embodiment in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

FIG. **1** illustrates an exemplary computer system environment that can be used in conjunction with the present invention. The exemplary environment includes a host computer **10** and an encryptor **12** at a Software Distribution Center (SDC), a satellite **14** for relaying transmitted software, a target computer **18**, and peripheral computers **20***a–n* which are coupled to the target computer **18**. As illustrated, the encryptor **12**, which encrypts software packages loaded on the host computer **10**, is separate from the host computer **10**. It is understood, however, that the encryptor **12** can be configured to be part of the host computer **10**.

In the exemplary computer environment of FIG. **1**, the target computer **18** contains a licensing software program **22** and a receiver/installer software program **24**, which are typically embodied on one or more program storage devices. A customer is licensed or otherwise authorized to use the broadcast service of the present invention. With such authorization, the customer receives the licensing program **22** and receiver/installer **24** program loads these programs onto the computer **18**, thereby enabling receipt of software transmitted in accordance with this invention. The licensing program **22** provides for the entry and tracking of strings for all software products purchased. The licensing program **22** also generates the unique computer identifier code (e.g., computer identification number) for the computer **10** that the installer/receiver program **24** runs. As described below, a customer provides this unique identification code whenever it places an order for a software package. The installer/receiver software program **24** enables the target computer **18** to receive software that is broadcasted, for example via satellite **14**, and to install the software onto the target computer **18**. The peripheral computers **20***a–n* also contain installer/receiver software.

The host computer **10** in FIG. **1** contains a library of stored software packages that customers may order. For each

software package, an encryption key is generated and the software package is encrypted using that encryption key. The encrypted software is then compressed and stored as one packet in a single compressed file. The compressed file also includes a packet which identifies the particular software product included in the file. As described in more detail below in connection with FIGS. **3** and **4**, each compressed file further includes a checksum packet which is generated at the time the software package is encrypted and compressed, enabling the target computer **18** to determine whether a software package is transmitted without error. A transmitting program on the host computer **10** contains information regarding the time at which each software package will be transmitted. The transmission and encryption of software are performed using techniques well-known in the art.

Referring now to FIG. **2**, a flow chart is illustrated detailing the steps which may be used to program the computer system of FIG. **1**, with the host computer **10** transmitting a software package to a target computer **18** through a broadcast medium, according to the present invention. Block **30** represents installation of a licensing program and receiver/installer program on the target computer **18**. As explained above, only users having this software on their computer receive software according to this invention; the encrypted software that is transmitted via satellite is otherwise unusable.

Blocks **32** and **34** represent the target computer **18** calling the SDC to order a particular software package. When the customer orders the software, the customer selects an option provided by the licensing program **22**, which produces the identification code of the target computer **18**. A customer gives his identification code to the operator receiving the software order at the SDC.

Block **36** represents the SDC encrypting the encryption key specific to the ordered software package. When the SDC receives the identification number of the target computer **18**, it produces a new key (e.g., an ASCII string) by encrypting the software encryption key for the ordered software package using the identification code of the target computer **18**. The purpose of the new key (i.e., the encrypted encryption key) is to lock a particular computer (e.g., the computer with the identification code used to encrypt the encryption key) to a particular software package. Instead of allowing any computer having the encryption key to listen to the broadcast and obtain software illegally, only the target computer **18** whose identification code is used to encrypt the software encryption key, accesses the transmitted software. As represented in block **38**, the SDC provides the new key to the target computer **18**. The SDC provides this string to the target computer **18**, for example, via fax or e-mail.

Block **40** represents the user of the target computer **18** entering the encrypted encryption key (i.e., new key) onto the target computer **18**. The installer/receiver program on the target computer **18** provides a user interface requesting the user to insert the new key provided by the SDC. The target computer **18** then stores this key.

Block **42** represents the SDC providing to the user of the target computer **18** the time and channel/frequency that the ordered software will be broadcast. Software products are broadcast from the SDC, for example, during scheduled timeslots. Preferably, popular products are broadcast at regular intervals, while relatively uncommon software products are periodically scheduled at the request of a user.

Block **44** represents transmission of the software from the SDC at the time and channel/frequency that was communicated to the user in block **42**. The software is broadcast as one compressed file via satellite. The bits of the file are pushed out on the line in a manner similar, for example, to transmission over TCP/IP or modem.

Block **46** represents the target computer receiving the software that is broadcast via satellite. At the time specified for transmission, users "tune in" with their receivers to the specified frequency or channel. As an option, the receiver could be set up to look for and download a specific list of ordered products, or specific versions of a product. For example, a user may want to always download new versions of products that were already installed so that the user always has the latest version of the software product.

Block **48** represents checking whether the transmission is successful. If the transmission is not successful, then the SDC provides the user with the relevant information regarding the next broadcast of the software. The target computer **18** then waits until that time that the software is retransmitted and subsequently receives the software as described in connection with block **46**. If the transmission is successful, then, as represented in block **50**, the target computer decrypts the encrypted encryption key using its identification code. Once this is performed, the target computer decrypts the software using the encryption key specific to the software product.

The decryption of the software, represented in block **52**, at the target computer **18** occurs as follows. The licensing program **22** generates the identification code of the target computer **18** which, in turn, is transmitted to the receiver/installer program on the target computer **18**. In the alternative, the receiver/installer program itself generates the identification code. The receiver/installer program **14** on the target computer **18** utilizes this identification number to decrypt the encrypted encryption key. The encryption key in turn enables the target computer **18** to decrypt (i.e., unlock) the transmitted software. This process occurs virtually simultaneously, thereby making it extremely difficult to identify the encryption key during the process. Consequently, only the target computer **18** can decrypt the software.

Block **54** represents determining whether there are other peripheral computers to receive the software. If there are no peripheral computers, the target computer **18**, as represented in blocks **56** and **58**, decompresses and installs the software onto the target computer **18**. Specifically, the installer/receiver program **24** invokes a routine to decrypt and decompress the temporary data file. This part of the installation copies files to the file destination with the appropriate permissions and ownership on the target computer **18**. Moreover, it kicks off any installation scripts required to complete the installation, configuration, or tuning specific for the application installed.

As depicted in blocks **60**, **62** and **64**, if there are other peripheral computers connected to the target computer **18**, then the target computer **18** distributes the software to the peripheral computers **20***a–n* which in turn decompress and install the software. As mentioned previously, each of the peripheral computers **20***a–n* has a receiver/installer software product to enable the decompression and installation of the software.

Referring to FIG. **3**, a compressed file for a software package is illustrated, having multiple packets. The compressed file, generally designated **70**, includes a header **72**, a compressed installable file packet **80**, a checksum packet **82**, and an end-product broadcast packet **84**. The header **72** includes a begin product broadcast packet **74**, a product

identification packet **76** and a version packet **78**. The begin product broadcast packet **74** signals the beginning of the transmission of a new software product. The product identification packet **76** identifies the software product that is transmitted. Version packet **78** identifies the particular version of the software product that is transmitted. For example, the version packet **78** notifies users when a software product upgrade is being transmitted. Other identifying information could also be included as part of the header **72**. For example, there could be a vendor header packet to enable a target computer to monitor for all software products that a particular vendor offers.

Packet **80** in the compressed file contains the compressed installable file. This includes the compressed software and the configuration scripts.

Packet **82** is the checksum packet. As described in more detail below with the respect to FIG. **4**, the checksum packet **82** enables the user at the target computer **18** to verify that the transmission is successful. The end product broadcast packet **84** notifies the target computer that the transmission is complete.

Referring now to FIG. **4**, a flow chart illustrates the steps for verifying that the software is successfully transmitted. Block **90** represents the transmission of the compressed software contained in the compressed installable file packet **80**. Block **92** represents transmission of the checksum packet **82**. The checksum packet **82** is generated at the SDC when the software product is initially compressed and stored at the send computer. Based on the contents of the file received at the target computer **18**, an algorithm contained in the receiver/installer program **24** generates a code or number based on the compressed file received at the target computer **18**. This number, which is generated using the same algorithm that generates the checksum when the software is compressed, is compared to the checksum packet **82**. If the transmission has been successful, the two numbers are the same. At block **94** of FIG. **4**, it is determined whether the transmission is successful by matching the checksum packet **82** with the code generated by the file copied to the target computer **18**. If the codes are the same, the target computer **18**, as depicted in block **96**, decrypts and decompresses the software and installs it on the target computer **18** or distributes it to other peripheral computers for decompression and installation. If the checksum packet **82** does not match the file that was received from the host computer **10**, then the target computer **18** deletes the file and waits for retransmission of the software product.

Unlike the prior art which provides transmission through phone lines, the present invention is not limited in the number of customers who can receive a given transmission. Each customer who orders a particular software product receives the software from the same broadcast. Therefore, theoretically a single broadcast transmits to an unlimited amount of users.

The foregoing description of the preferred embodiment of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible without departing from the scope of the invention as defined by the claims appended hereto.

What is claimed is:

1. A method of distributing software encrypted by a software encryption key to a target computer, comprising the steps of:

receiving a number unique to the target computer;

encrypting the software encryption key using the number unique to the target computer unique number;

transmitting the encrypted software encryption key to the target computer;

communicating a timeslot and channel for broadcasting the encrypted software to the target computer; and

broadcasting a message at the communicated timeslot and channel, the message comprising the encrypted software.

2. The method of claim **1**, wherein the broadcast timeslot is periodically scheduled according to a user request.

3. The method of claim **1**, wherein the transmission timeslot is regularly scheduled.

4. The method of claim **1**, wherein:

the message further comprises a version packet notifying the target computer that an upgraded version of the software is being transmitted and enabling the target computer to download upgraded versions of software already installed on the target computer; and

the method further comprises the step of decrypting the software using the target computer unique number when the version indicator indicates that an upgraded version of software already installed on the target computer is being broadcast.

5. The method of claim **1**, wherein:

the message further comprises a vendor header packet enabling the target computer to monitor broadcasts for software products offered by a vendor.

6. The method of claim **1**, further comprising the steps of:

decrypting the software in the target computer using the target computer unique number;

determining if there are peripheral computers coupled to the target computer to receive the decrypted software; and

distributing the decrypted software to the peripheral computers when there are peripheral computers coupled to the target computer to receive the decrypted software.

7. The method of claim **1**, wherein the message further comprises a checksum packet for determining if the target computer received the encrypted software without error, and the method further comprises the steps of:

receiving the checksum packet in the target computer;

processing the checksum packet in the target computer to determine if the encrypted software was received without error;

providing the target computer information regarding a next broadcast of the software when the encrypted software was not received without error; and

waiting for retransmission of the encrypted software.

8. An apparatus for distributing software encrypted by a software encryption key to a target computer, comprising:

means for receiving a number unique to the target computer;

means for encrypting the software encryption key using the number unique to the target computer unique number;

means for transmitting the encrypted software encryption key to the target computer;

means for communicating a timeslot and channel for broadcasting the encrypted software to the target computer; and

means for broadcasting a message at the communicated timeslot and channel, the message comprising the encrypted software.

9. The apparatus of claim 8, wherein the means for providing a timeslot and channel for broadcasting the encrypted software to the target computer comprises means for periodically scheduling the timeslot according to a user request.

10. The apparatus of claim 8, wherein the means for providing a timeslot and channel for broadcasting the encrypted software to the target computer comprises means for regularly scheduling the transmission time.

11. The apparatus of claim 8, wherein:

the message further comprises a version packet notifying the target computer that an upgraded version of the software is being transmitted and enabling the target computer to download upgraded versions of software already installed on the target computer; and

the apparatus further comprises means for decrypting the software using the target computer unique number when the version indicator indicates that an upgraded version of software already installed on the target computer is being broadcast.

12. The apparatus of claim 8,wherein the message further comprises a vendor header packet enabling the target computer to monitor broadcasts for software products offered by a vendor.

13. The apparatus of claim 8, further comprising:

means for decrypting the software in the target computer using the target computer unique number;

means for determining if there are peripheral computers coupled to the target computer to receive the decrypted software; and

means for distributing the decrypted software to the peripheral computers when there are peripheral computers coupled to the target computer to receive the decrypted software.

14. The apparatus of claim 8, wherein the message further comprises a checksum packet for determining if the target computer received the encrypted software without error, and the apparatus further comprises:

means for receiving the checksum packet in the target computer;

means for processing the checksum packet in the target computer to determine if the encrypted software was received without error;

providing the target computer information regarding a next broadcast of the software when the encrypted software was not received without error; and

waiting for retransmission of the encrypted software.

15. A program storage device, readable by computer, tangibly embodying one or more programs of instructions executable by the computer to perform method steps of distributing software encrypted by a software encryption key to a target computer, the method steps comprising the steps of:

receiving a number unique to the target computer;

encrypting the software encryption key using the number unique to the target computer unique number;

transmitting the encrypted software encryption key to the target computer;

communicating a timeslot and channel for broadcasting the encrypted software to the target computer; and

broadcasting a message at the communicated timeslot and channel, the message comprising the encrypted software.

16. The program storage device of claim 15, wherein the broadcast timeslot is periodically scheduled according to a user request.

17. The program storage device of claim 15, wherein the transmission timeslot is regularly scheduled.

18. The program storage device of claim 15, wherein:

the message further comprises a version packet notifying the target computer that an upgraded version of the software is being transmitted and enabling the target computer to download upgraded versions of software already installed on the target computer; and

the method steps further comprises the method step of decrypting the software using the target computer unique number when the version indicator indicates that an upgraded version of software already installed on the target computer is being broadcast.

19. The program storage device of claim 15, wherein:

the message further comprises a vendor header packet enabling the target computer to monitor broadcasts for software products offered by a vendor.

20. The program storage device of claim 15, wherein the method steps further comprise the method steps of:

decrypting the software in the target computer using the target computer unique number;

determining if there are peripheral computers coupled to the target computer to receive the decrypted software; and

distributing the decrypted software to the peripheral computers when there are peripheral computers coupled to the target computer to receive the decrypted software.

21. The program storage device of claim 15, wherein the message further comprises a checksum packet for determining if the target computer received the encrypted software without error, and the method steps further comprise the method steps of:

receiving the checksum packet in the target computer;

processing the checksum packet in the target computer to determine if the encrypted software was received without error;

providing the target computer information regarding a next broadcast of the software when the encrypted software was not received without error; and

waiting for retransmission of the encrypted software.

* * * * *

United States Patent [19]

Durden et al.

[11] **Patent Number: 5,003,384**

[45] **Date of Patent: Mar. 26, 1991**

[54] **SET-TOP INTERFACE TRANSACTIONS IN AN IMPULSE PAY PER VIEW TELEVISION SYSTEM**

[75] Inventors: **Gregory S. Durden**, Jonesboro; **Ray T. Haman, Jr.**; **Scott L. Hamilton**, both of Duluth; **Richard B. Frederick**; **David J. Naddor**, both of Doraville; **Randolph J. Schaubs**, Stone Mountain, all of Ga.

[73] Assignee: **Scientific Atlanta, Inc.**, Atlanta, Ga.

[21] Appl. No.: **176,776**

[22] Filed: **Apr. 1, 1988**

[51] **Int. Cl.**[5] **H04N 7/167**; H04N 7/10

[52] **U.S. Cl.** **358/84**; 455/2; 358/349; 358/86

[58] **Field of Search** 358/84, 86, 349; 379/92; 455/2, 4, 5; 380/20

[56] **References Cited**

U.S. PATENT DOCUMENTS

Re. 31,735	11/1984	Davidson	380/20 X
3,790,700	2/1974	Callais et al.	358/84
3,886,302	5/1975	Kosco	380/20
4,277,651	7/1981	Fisher, II et al. .	
4,484,217	11/1984	Block et al.	358/84
4,536,791	8/1985	Campbell et al.	358/86 X
4,584,602	4/1986	Nakagawa	358/84
4,630,108	12/1986	Gomersall	358/84
4,686,564	8/1987	Masuko et al.	358/86
4,710,955	12/1987	Kauffman	358/84 X
4,739,510	4/1988	Jeffers	358/84
4,792,284	12/1988	Nussrallah et al.	358/86

Primary Examiner—Jin F. Ng
Assistant Examiner—Stella L. Woo
Attorney, Agent, or Firm—Banner, Birch, McKie & Beckett

[57] **ABSTRACT**

An impulse pay-per-view system wherein a number of downloadable transactions may be utilized to effect increased control and diversity is disclosed. A preview time/free transaction permits a system subscriber to view portions of a pay-per-view event without purchasing the event. A security transaction sent while a pay-per-view event is active may be used to prevent a subscriber from receiving events which have not been purchased. A telephone number transaction may include a special character representing an instruction to pause between the dialing of selected digits. A viewer statistic transaction may be use dot instruct a subscriber to record the channel he is viewing. Finally, a subscriber may pre-buy pay-preview events in order to facilitate VCR recording of the events while he is asleep or away from home.

48 Claims, 3 Drawing Sheets

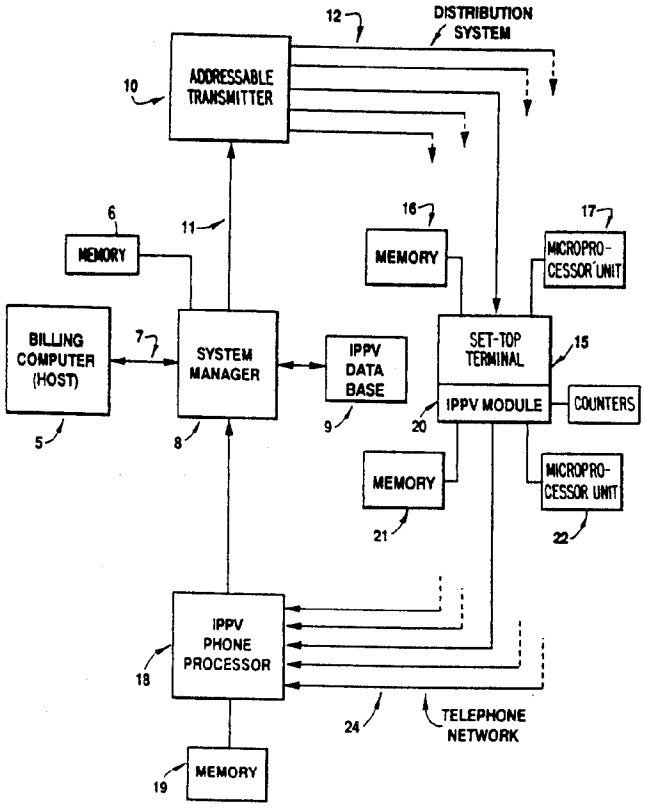


FIG. 1

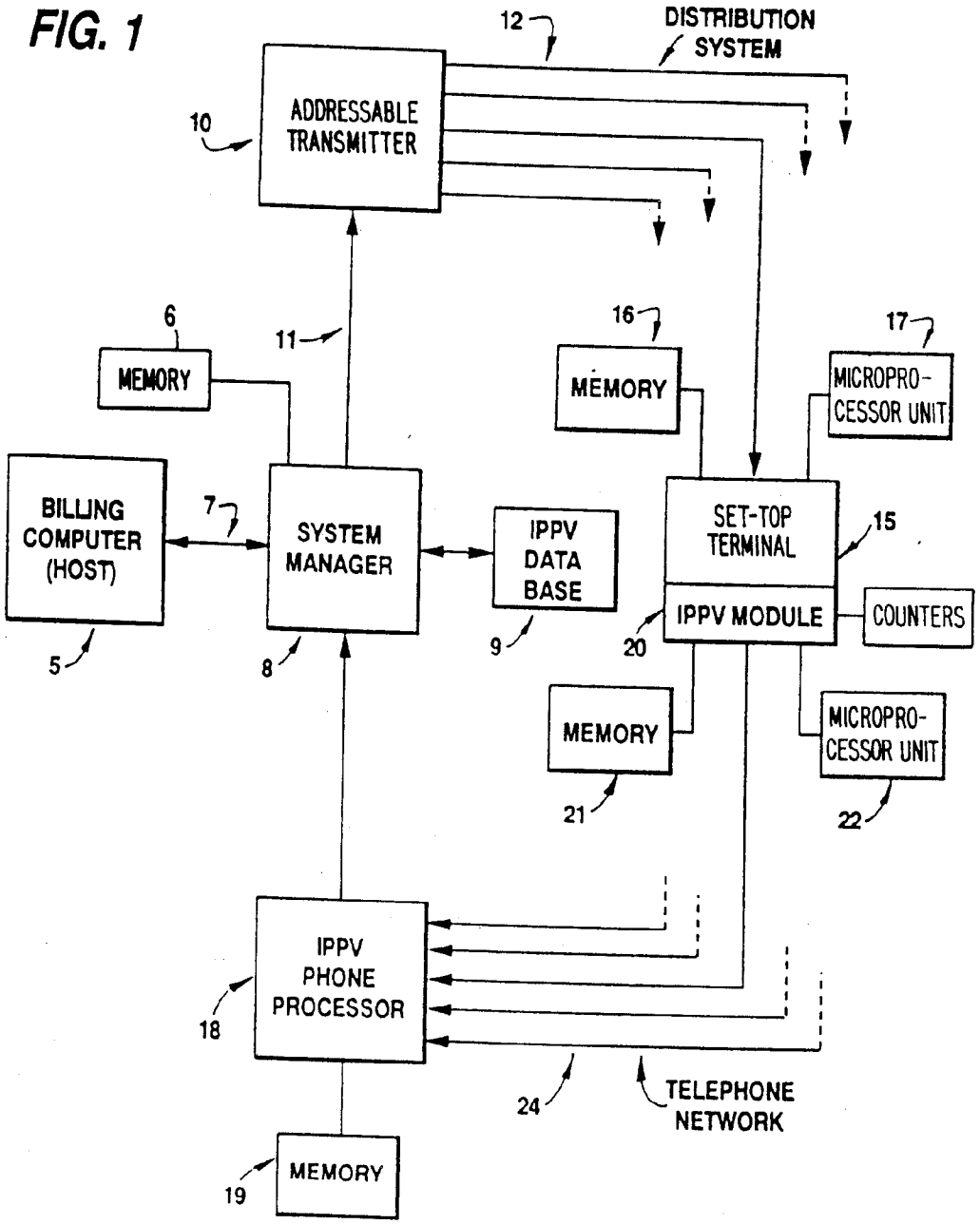

12
DISTRIBUTION SYSTEM
ADDRESSABLE TRANSMITTER
10
6
MEMORY
11
16
MEMORY
17
MICROPRO-CESSOR UNIT
BILLING COMPUTER (HOST)
7
SYSTEM MANAGER
IPPV DATA BASE
SET-TOP TERMINAL
15
IPPV MODULE
COUNTERS
5
8
9
20
MEMORY
MICROPRO-CESSOR UNIT
21
22
IPPV PHONE PROCESSOR
18
24
TELEPHONE NETWORK
MEMORY
19

FIG. 2

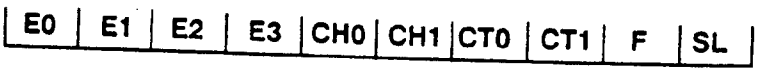


FIG. 4

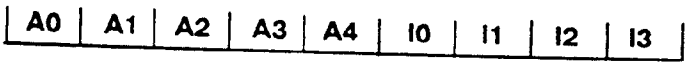


FIG. 5a

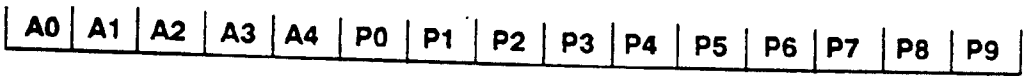


FIG. 5b

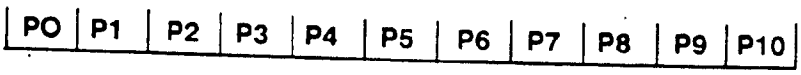


FIG. 6

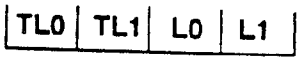
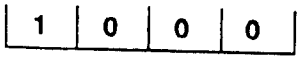


FIG. 7



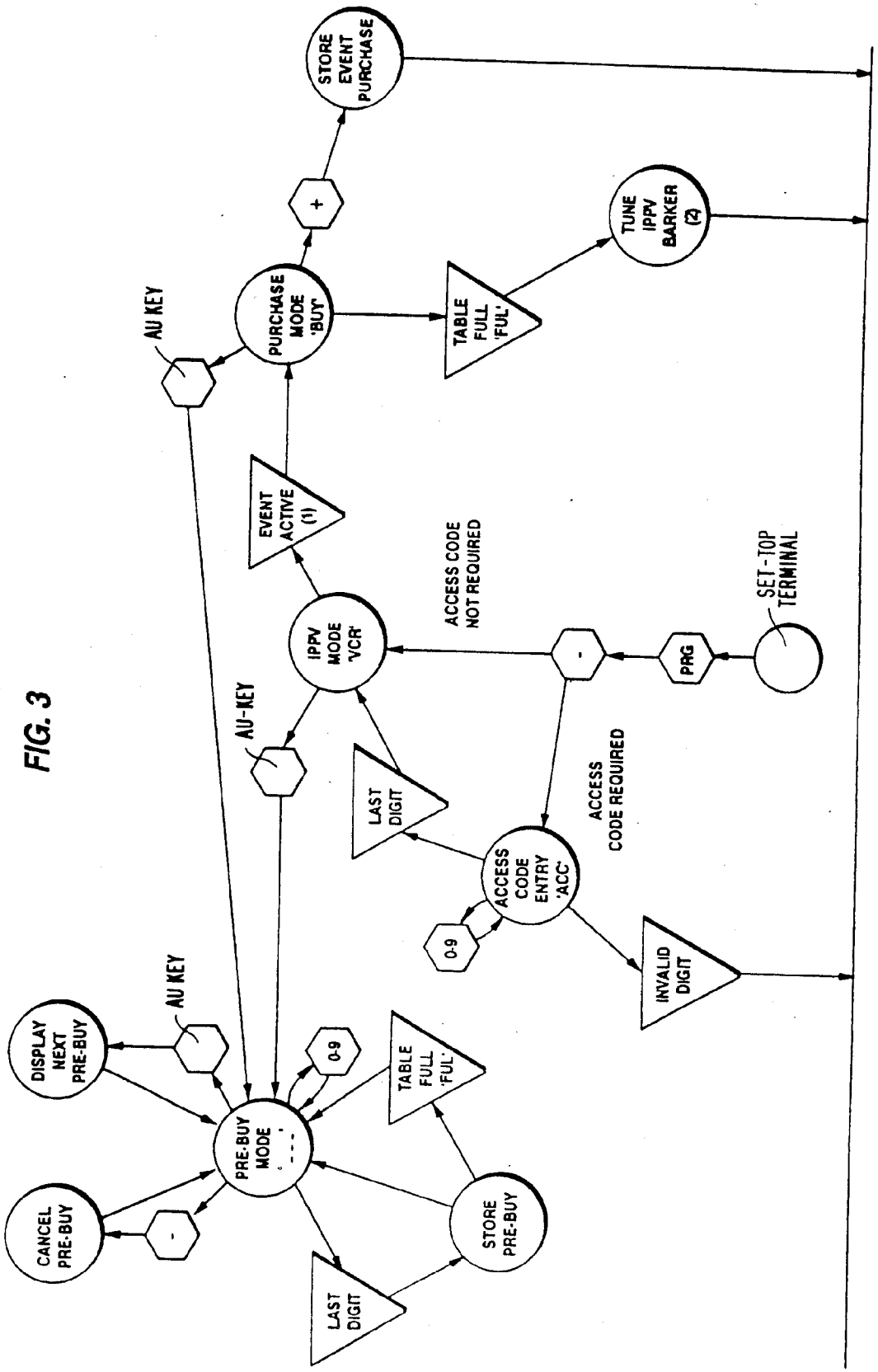

STORE EVENT PURCHASE
+
TUNE IPPV BARKER (2)
AU KEY
PURCHASE MODE 'BUY'
TABLE FULL 'FUL'
EVENT ACTIVE (1)
ACCESS CODE NOT REQUIRED
SET-TOP TERMINAL
IPPV MODE 'VCR'
-
PRG
AU-KEY
LAST DIGIT
ACCESS CODE REQUIRED
ACCESS CODE ENTRY 'ACC'
0-9
INVALID DIGIT
AU KEY
DISPLAY NEXT PRE-BUY
0-9
TABLE FULL 'FUL'
PRE-BUY MODE
CANCEL PRE-BUY
-
STORE PRE-BUY
LAST DIGIT

FIG. 3

SET-TOP INTERFACE TRANSACTIONS IN AN IMPULSE PAY PER VIEW TELEVISION SYSTEM

BACKGROUND OF THE INVENTION

This invention relates generally to the field of cable television systems and, more particularly, to a system for reporting back to a centrally located office, the viewing of certain premium programming for which a subscriber is billed in addition to his regular monthly subscription fee. This practice is popularly referred to as "pay-per-view" (PPV). More specifically, the subject invention relates to the most desirable type of PPV, known as "impulse pay-per-view" (IPPV). An IPPV system allows a subscriber to order a program at the last minute.

Early PPV systems worked with one way addressable set-top terminals (STT). A subscriber who wished to order a PPV event did so by calling an operator and orally placing his order. The operator entered the order into a computer, which then transmitted authorization to the subscriber's set-top terminal.

This system suffers from the requirement of using the telephone and a human operator. This increases the cost of handling PPV requests, and effectively eliminates IPPV as a viable service since only a limited number of people are able to call in during the last minutes before a program begins. Therefore, the majority of people desiring to view a program must order it long before it begins.

Some prior art systems exist which purport to solve the IPPV problem. One such system employs a two-way cable television (CATV) plant, in which the set-top terminal may be equipped for transmitting a signal back to the headend ("upstream transmission") on a suitable frequency, such as between 5 MHz and 30 MHz. The terminal transmits information as to what programs are being or have been viewed to a computer at the headend. This system suffers from the fact that no protocols have been developed which operate efficiently in an enviroment of an exceedingly large number of set-top terminals who "speak" very little, but who must be serviced quickly when they do speak. Further, two-way CATV plants have proved difficult to maintain with adequate integrity to permit reliable return transmission, and the cost of the plant is excessive compared to the revenues to be gained from IPPV.

Another system uses credits downloaded to the terminal, and then makes deductions against the credits when a program is viewed. At the end of the month, certain alpha-numeric characters are displayed, indicating programs viewed. The subscriber writes these characters on a card which is mailed to the CATV operator. This system suffers from excessive delay in reporting programs watched, a limited number of programs which can be viewed (due to the limited number of characters a subscriber can be expected to write down), and the possibility of unrecoverable errors in transcription.

A similar system is disclosed in U.S. Pat. No. 4,484,217 to Block. In this system, credits are downloaded to the terminal and deductions are made when programs are viewed. An indicator informs the subscriber that the stored credit has expired or is low. The headend office, upon receipt of payment, will add credits to the subscriber's terminal and the indicator will be extinguished. In this system the subscriber must pay in advance and may miss programs due to delay in crediting his account.

An impulse pay-per-view system is disclosed in commonly assigned U.S. Pat. No. 4,792,848, herein incorporated by reference.

SUMMARY OF THE INVENTION

It is an object of the present invention to effect increased control and enhance the diversity of an impulse pay-per-view system.

This object maybe achieved in a control apparatus for an individual subscriber in a cable television system which distributes a television signal from a headend office to a plurality of subscribers, the television signal including a plurality of channels and a plurality of downloadable transactions. At least one of the channels carries pay-per-view events. The control apparatus includes a receiver for receiving television signals, a detector for detecting any downloadable transactions contained in the television signals which are addressed to the corresponding subscriber, and a microprocessor for processing the transactions detected by the detector. A selector is responsive to subscriber supplied signals for selecting pay-per-view events carried over one of the plurality of channels. A memory is coupled to the microprocessor and the selector for storing billing information regarding selected pay-per-view events. A transmitter is coupled to the storage means for transmitting the stored billing information. The apparatus further includes a device responsive to a downloadable transaction which permits the subscriber to receive a selected pay-per-view event for a predetermined preview time period before the subscriber must supply a signal to generate billing information. The apparatus includes another device responsive to the downloadable transaction which further permits the subscriber to receive the event for a predetermined free time period time before supplying a signal to generate billing information.

The present invention also concerns a control apparatus for downloading transactions to a plurality of subscribers. This apparatus includes a downloadable transmitter which transmits the downloadable transactions to the plurality of subscribers. A processor processes instructions from a system operator. A preview time generator, responsive to the system operator, generates a downloadable transaction to permit the plurality of subscribers to receive an event for a predetermined preview time period before requiring the subscribers to purchase an event. The apparatus also includes a free time generator responsive to the system operator which generates a downloadable transaction which permits the plurality of subscribers to further receive the event for a predetermined free time period before requiring the subscribers to purchase the event.

The present invention is further concerned with a method of pre-buying a selected event which is being shown on one of a plurality of channels. First, a predetermined key sequence is actuated, the key sequence generating information corresponding to a selected event on one of the plurality of channels. The information thus generated is then transmitted to an event module. The generated information is stored in a memory associated with the event module and subsequently compared with information contained in a downloadable transaction. A subscriber is authorized to view the selected event if the stored information matches the information contained in the downloadable transaction.

Billing information corresponding to the selected event is generated and stored in the memory.

The present invention is further concerned with a method of instructing a plurality of subscriber modules to report over a public telephone network billing information associated with the viewing of selected events on the plurality of channels. A transaction is downloaded to the subscriber module which includes a telephone number corresponding to a storage means for storing billing information associated with the plurality of subscribers. Included in this transaction are instructions adapted to effect a predetermined delay period between the dialing of selected digits.

The present invention is further concerned with a method of transmitting billing information associated with the viewing of selected events on a plurality of channels from a subscriber module to the headend office. A telephone number is loaded in a memory associated with the module in response to information contained in a first transaction downloaded from the headend office. This telephone number is dialed in response to instructions contained in a second transaction downloaded from the headend office. In response to instructions contained in the first downloaded transaction, pauses of a predetermined period are inserted between the dialing of selected digits.

The present invention is also concerned with a control apparatus for an individual subscriber designed to protect against theft of services. The apparatus includes a receiver for receiving television signals, a detector for detecting downloadable transactions in the television signal which are addressed to the corresponding subscriber, and a microprocessor for processing transactions detected by the detector. A selector is responsive to subscriber supplied signals to select events carried over the plurality of channels. A memory is coupled to the microprocessor and the selector for storing billing information regarding the selected event. A transmitter coupled to the memory transmits the stored billing information over a telephone network. A security counter responsive to a downloadable transaction from the headend which is sent only while an event is active prevents the subscriber from receiving an event unless the transaction is received.

The present invention is further concerned with a control apparatus for downloading transactions to a plurality of subscribers. The apparatus includes a downloadable transmitter for transmitting the downloadable transactions to a plurality of subscribers and a processor for processing instructions from a system operator. A security generator responsive to the system operator generates a downloadable transaction only while an event is active which prevents a subscriber from receiving an event unless the transaction is received.

The present invention is further concerned with a method of generating viewer statistics. Initially, a first downloadable transaction is addressed to at least one subscriber module with instructions to store the channel number corresponding to the channel being watched by the subscriber in an associated storage device at the moment the first downloadable transaction is received. A second downloadable transaction is addressed to the subscriber module with instructions to the module to initiate telephone communication with the headend office. The second transaction includes instructions to the module to transfer the stored channel number to the headend and to write a predetermined character in the memory as an indication that the channel number has been transferred.

Finally, the present invention is concerned with a method of processing viewer statistics at a subscriber module. Initially, a first downloaded transaction is received with instructions to store the channel number corresponding to the channel being watched by the subscriber. This channel number is stored in a storage device. In response to a second downloaded transaction, the module dials into the public switched telephone network to establish communication with the headend office. The stored channel number is transferred to the headend office and a predetermined character is written to the storage device as indication that the channel number has been transferred.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete appreciation of the invention and many of the attendant advantages thereof will be readily obtained as the invention becomes better understood by reference to the following detailed description when considered in connection with the accompanying drawings, wherein:

FIG. **1** is a block diagram representing the overall configuration of the impulse pay-per-view system according to the present invention.

FIG. **2** is a schematic representation of the bit patterns in an authorization transaction.

FIG. **3** is a state diagram illustrating the method of pre-buying an IPPV event from a hand-held remote or set-top.

FIG. **4** is a schematic representation of bit patterns in a pre-buy transaction downloaded by the system manager.

FIG. **5***a* and **5***b* are schematic representations of the bit patterns in an addressable and a global load telephone number transaction, respectively.

FIG. **6** is a schematic representation of bit patterns in a data request transaction.

FIG. **7** is a schematic representation of a bit patterns in a viewing statistic transaction.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

An overview of the addressable impulse pay-per-view system according to the present invention will be given with respect to FIG. **1**. In what follows, hexadecimal notation 0-F will be used to denote data values. The system includes a billing computer or host **5** which comprises an essential part of an addressable cable system having impulse pay-per-view capability. Billing computer **5** records and maintains records for each cable subscriber. These records may contain information such as the subscriber's name, address and telephone number, the type of equipment the subscriber has in his possession, and which pay services the subscriber is authorized to view. With particular regard to both impulse pay-per-view and pay-per-view programming, billing computer **5** functions to control IPPV service, maintain IPPV access codes, control IPPV event billing, and maintain PPV event and preview definitions. Typically, the cable operator owns the billing computer, leases the equipment from a vendor who specializes in this type of equipment, or shares computer time on a machine owned by one of these billing vendors.

Billing computer **5** is interfaced to a system manager **8**. System manager **8** controls the addressable cable system. Typically located at or near the billing com-

puter 5, system manager 8 maintains a list of all the addressable set-top terminals in the cable system as well as those services which each terminal is authorized to receive. System manager 8 also defines and maintains the parameters selected by the cable operator for each system. These parameters may include the frequencies associated with each channel in the system, which channels are being scrambled, the security features of the system, and the system time. Additionally, system manager 8 is responsible for the authorization and deauthorization of pay-per-view events in the system.

A computer of system manager 8 will have a disk and controller dedicated to the storage of IPPV information. A memory resident program of system manager 8 will read the IPPV transactions, uploaded from the IPPV modules in the system. Each IPPV transaction will be stored in data base 9 until it is retrieved by billing computer 5. An operator of the system manager will have the ability to access the IPPV data base 9 and display the event identifiers (IDs) of the events that have been purchased by the converter. System manager 8 will upload the transaction data to billing computer 5 in response to an Initialize IPPV Upload command. The data returned to the billing computer will be grouped by the identification of a particular set-top terminal or converter, that is, all event IDs associated with a particular converter serial number will be returned together.

The system manager will also maintain a table of viewing statistics in memory 6 and on the IPPV disk. System manager 8 will further be able to direct all IPPV-equipped set-top terminals to record the channel to which each is tuned by transmitting a viewing statistic transaction (discussed in detail below) via addressable transmitter (ATX) 10. This information is recorded in the IPPV module's memory and is transmitted along with the IPPV transaction data during the next interrogation of the module. Thus, each time an IPPV transaction is received from an IPPV module, a table location corresponding to the recorded channel number contained in the IPPV transaction will be incremented. A value of '00', for example, may be used to indicate that a set-top terminal was off when the viewing statistic transaction was transmitted by the system manager. A value of 'FF' may be used to indicate that a set-top has already transferred its channel information to the system manager. The system manager will thus have the capability to print a summary report of viewing statistics for each channel. The system manager will also clear the table containing the viewing statistics whenever a new viewing statistic transaction is transmitted.

Both the IPPV access code and IPPV service code will be maintained for each IPPV equipped converter in the system manager converter data base. System Manager 8 will also maintain data elements for each headend in its data base. These data elements may include the telephone number for each IPPV module transfer, IPPV "HELP" barker channel, IPPV "FULL" barker channel, call back limit, security nibble, host time out value, IPPV time out-counter, transfer key, and IPPV ID range. Briefly, the "FULL" barker channel refers to an STT Status which specifies the channel to be tuned by the IPPV Module when an attempt is made to purchase an IPPV event and there is no room in the IPPV Module's event purchase table to store the event purchase data. The security nibble refers to a transaction used to maintain the security of the IPPV return data path. The value of this field may be sent to the IPPV Module by the ATX during a define telephone number transaction described below. Each time the IPPV Module calls the IPPV phone processor, this value is included in the data returned. The system manager can then compare the value obtained from the IPPV Module with the value being transmitted by the ATX to detect possible data security violations. The host timeout refers to a transaction which specifies the amount of time an IPPV Module will wait for a response from the IPPV phone processor. If no response is received, the IPPV Module will consider the call to be unsuccessful and retry at a later time. Transfer key refers to a transaction which is used to control the number of IPPV modules that will call the IPPV phone processor. The range of valid values may be, for example, 0 to 7 and F, where F indicates a global request and any other value causes only those converters to call for which the least significant 3 bits of the digital address matches the value of the transfer key. The IPPV ID range refers to a transaction which specifies the number of digits to be used by the IPPV module when entering or displaying the IPPV Event ID for an IPPV Pre-Buy.

System manager 8 will control IPPV module telephone usage by transmitting an IPPV request data transaction to ATX 10. System Manager 8 will accept host link commands to allow the host computer 5 to add, modify and delete PPV event definitions. An operator of the system manager will also be able to display and update this data. System manager 8 will schedule the global authorization and deauthorization of PPV events. For each PPV event, the event ID, free time and preview time (discussed in detail below), purchase window and event slot will be transmitted along with the authorizations.

The addressable transmitter (ATX) 10 is a device used to receive commands generated by system manager 8 and transmit them on a dedicated data channel in the cable system in a format recognizable by the addressable set-top terminals. Typically, one ATX is located at each cable headend. ATX 10 will receive IPPV transactions from system manager 8 and format the appropriate commands for transmission to the IPPV equipped converters.

Each subscriber in the addressable cable system is provided with a set-top terminal (STT) 15 by the cable operator as schematically indicated in FIG. 1. STT 15 allows the subscriber to tune and descramble the services that he has requested from the cable system operator. Each STT 15 contains a unique identifier, or address, for example, a manufacturer's serial number, which allows the cable operator to send commands via ATX 10 to an individual STT. These commands are called "addressable" commands. The STT's are also capable of receiving "global" commands heard by all the STT's in a cable system. Those subscribers who are authorized to purchase impulse pay-per-view events are issued set-top terminals which have been equipped with an IPPV module 20. Module 20 allows the subscriber to authorize his STT to receive a pay-per-view event, store the data associated with the purchase of that event in memory 21, and transmit that stored data to the cable operator via the telephone network 24. The subscriber is then billed for the events that he has purchased. IPPV module 20 receives the IPPV transactions from ATX 10 via distribution system 12 and stores the IPPV parameters in non-volatile memory (NVM 21). The IPPV module also functions to authorize IPPV events and record IPPV activity in the NVM 21. Module 20 transfers IPPV data to the system manager 8 via phone pro-

cessor 18 when a "request IPPV data" command is received. Upon successful transmission, the IPPV module will clear the events watched table maintained in NVM 21.

The interfaces between these various components will now be discussed. Billing Computer or Host 5 sends data to system manager 8 via a serial interface schematically indicated at 7. Host 5 acts as the master and system manager 8 acts as a slave device. That is, the Host 5 sends a command to the system manager 8, the system manager processes the command and sends a response to the Host, and the sequence is repeated. The system manager may not initiate a conversation, but can only respond to commands from the Host. There are generally three types of commands associated with the IPPV System of the present invention which are sent via interface 7. The first type includes set-top terminal commands. These commands are used to configure a subscriber's STT for Impulse PPV. They allow the Host 5 to authorize a subscriber for IPPV purchases and define the "access code" that must be entered by the subscriber in order to purchase an IPPV event. The second type of commands includes pay-per-view commands. These commands are used to define the characteristics of the pay-per-view events being shown in the system, including the start and stop times, event ID, channel, amount of free time, and the period in which each event is available for purchase. System Manager 8 is thus responsible for authorizing and deauthorizing the PPV events at the appropriate times and controlling the characteristics of each event. Finally, the third type of commands include upload commands. These commands are used by the Host 5 to retrieve the data that has been collected by the system manager 8 from the plurality of Impulse PPV Modules 20 via phone processor 18. This data is transmitted to the system manager by the IPPV module 20 via the telephone network 24 and contains a record of which PPV events have been purchased by each subscriber. The Host uses this data to bill the subscribers for each PPV event purchased.

The system manager 8 transmits data to the ATX 10 via a serial interface in which the system manager 8 acts as the master and the ATX 10 acts as the slave device. There are generally four types of commands associated with the IPPV System which are sent via this interface. First, set top commands are sent which are used to configure an STT for Impulse PPV. They allow the system manager to authorize a subscriber for IPPV purchases, and define the "access code" that must be entered in order to purchase an IPPV event. Authorize Pay-Per-View Event commands are used by the system manager to start PPV events in the system and to define the characteristics of the events, including the event ID, channel, amount of free time and the purchase window. Purchase window refers the period in which a subscriber is allowed to purchase the event. The purchase window always begins at the start time of an event. The authorize IPPV/PPV event transaction authorizes the channel corresponding to a PPV event in all converters in which the specified PPV identification code has been loaded. The transaction also authorizes the channel corresponding to an IPPV event in all converters with IPPV modules that have requested the event. It also defines the amount of free time for the event. Deauthorize Pay-Per-View Event commands are used by the system manager to stop PPV events in the system. The deauthorize/purge IPPV/PPV transaction deauthorizes the channel corresponding to a PPV event in all converters in which the specified PPV identification code has been loaded. In addition, the PPV identification code is purged from every converter. This transaction also deauthorizes the channel corresponding to an IPPV event. Finally, IPPV System Parameter commands are used to configure the IPPV modules for IPPV operation. They allow the cable operator to define IPPV help and barker channels, the telephone number to be used by the IPPV modules to transfer data to the system manager, and to define the method used by the IPPV modules to determine when to phone the system manager and transfer the data associated with each PPV event purchased. The request data from IPPV module transaction directs an IPPV module to initiate a phone call to the system manager to transmit IPPV data stored in the module. The request viewing statistics transaction directs an IPPV module to store the channel to which it is tuned when this transaction is received. The load telephone number transaction defines the telephone number to be used by the IPPV module to transfer data to the system manager. The IPPV event load transaction loads a PPV event ID into an IPPV enabled converter.

ATX 10 transmits data to the IPPV module 20 via a dedicated FM data channel in the cable television distribution system 12. This channel, known as the "data carrier", is used to transmit both "addressable" commands intended for a particular IPPV module and "global" commands intended for all IPPV modules in the system. Alternatively, the transaction data may be inserted into the television signal itself, such as in the vertical blanking interval. Each addressable STT contains a receiver that is listening to the commands sent down this data channel. This data is available to the IPPV module in every IPPV-equipped STT. (An IPPV module may be designed as a plug-in module for reception into a set top terminal at an appropriate interface. Alternatively, the IPPV module circuitry may be incorporated within the circuitry of the set top terminal, thereby promoting efficiencies of operation. For example, the microprocessor of the STT may be appropriately designed to control both a billing memory normally associated with the module and an IPPV event memory of the STT). This is a one way data path, i.e., the IPPV module 20 cannot use this data channel to send data back to the ATX 10. Also, this data channel is available to the STT at all times as long as the incoming cable is connected, no matter what video/audio channel the STT is tuned to. The commands sent via this interface are similar to those sent via the system manager-ATX interface. Set-top commands are used to configure an IPPV-equipped STT for Impulse PPV purchases. They enable the module for IPPV purchases and define the "buy code" that must be entered in order to purchase an IPPV event. Authorize Pay-Per-View event commands are sent from the ATX 10 to start PPV events in the system and to define the characteristics of the events including the event ID, channel, amount of free time, and the purchase window. Each of these commands must be transmitted at the beginning of each PPV event, as well as during the event. Deauthorize Pay-Per-View Events commands are used to stop PPV events in the system. These commands must be transmitted at the conclusion of each PPV event. IPPV System Parameter Commands are used to configure the IPPV modules for IPPV operation. They allow the cable operator to define IPPV help and barker channels, the telephone number to be used by the IPPV

modules to transfer data to the system manager, and to define the method use by the IPPV modules to determine when to phone the system manager and transfer the data associated with each PPV event purchased. Finally, Time of Day commands define the date and time to the system.

The IPPV module is interfaced to the system manager via an IPPV phone processor **18**. The telephone network interface between each IPPV module and the system manager will be used for a single transaction, i.e. to upload viewing statistics and IPPV program purchases to the system manager. The IPPV modules **20** transmit data to the system manager **8** via the telephone network **24** and phone processor **18**. The system manager periodically instructs the IPPV modules via the ATX **10**, to use the subscriber's telephone line to send data associated with the IPPV events purchased by the subscriber with the IPPV module. For each event purchase stored in the module, the event ID and time of purchase are transferred to the system manager. The IPPV module is connected via the telephone network **24** with a phone processor **18** which manages calls initiated by set-top terminals equipped with IPPV capability. It stores the IPPV event information contained in these calls in nonvolatile memory **19** until the information can be uploaded to the system manager or other control computer. After receiving the IPPV information the phone processor **18** transmits the proper security codes to clear IPPV events from the set-top terminals.

The phone processor **18** functions as a peripheral device to the system manager **8**. Each phone processor monitors up to eight separate telephone lines for incoming calls. As the calls arrive, the phone processor answers the proper telephone line and transmits a transaction to the set-top terminal. Upon successful reception of that transaction, the set-top terminal then transmits all stored event information to the phone processor. If the information is received error-free, the phone processor transmits a transaction which the set-top terminal must decode to clear the event information. After the event data is received from the set-top terminal, it is stored in a non-volatile memory array in phone processor **18** which preserves the information through any loss of power.

Periodically, the phone processor **18** attempts to upload to the system manager **8** with a buffer packet message. The buffer packet message contains the stored event information and/or several status information fields. If the phone processor is unable to transfer the information, due to power loss or serial link failure, the phone processor non-volatile memory **19** fills up and the phone processor **18** will not accept any additional telephone calls until memory space becomes available.

When the system manager receives the buffer packet from the phone processor, it verifies a checksum. If the checksum is incorrect, a command is sent to the phone processor which causes the buffer packet to be retransmitted. If the checksum is correct, the associated data is permanently removed from the memory of phone processor **18**. The system manager must save its data on a disk or in some non-volatile memory before sending the command to clear the data from the phone processor in order to avoid the irrevocable data loss in the case of power failure.

The present invention is specifically concerned with a number of unique set-top transactions which may be sent via ATX **10** to effect increased control and enhance the diversity of an impulse pay-per-view system. A number of these transactions were mentioned briefly above and will be discussed in greater detail below.

The Authorize IPPV Event transaction will be discussed first with reference to FIG. 2. FIG. 2 illustrates bit patterns which may be included in this transaction. E0–E3 represent the IPPV event ID. CH0–CH1 represent the associated channel. F represents the free time. CT0–CT1 represent a counter reset value. SL represents a slot value. An additional bit pattern (not shown) may be included to specify the purchase window. As noted above, this authorization transaction authorizes the channel corresponding to a PPV event in all converters in which the specified PPV identification code has been previously loaded, as well as authorizing the channel corresponding to an IPPV event in all converters with IPPV modules that have requested the event. This transaction, as indicated, includes a field that may be used to control what are called preview time and free time. Preview time is defined as a period or window of time at the beginning of an event during which the event or a preview of a future event may be watched without being purchased. Free time is defined as a cumulative length of time during an event that the event may be watched without being purchased. Free time may be used at any time during the event. In a preferred embodiment, preview time may range from 0 to 255 minutes while free time may range from 0 to 14 minutes. For example, suppose the IPPV event consists of a movie with a starting time of 8:00 p.m. and an ending time of 10:00 p.m. The system operator may designate a block of time, say from 8:00–8:30, during which the movie may be viewed without the viewer having to purchase the movie. This would constitute the preview time. The system operator may elect to permit additional viewing of the movie for a total of fourteen minutes during the event time. Thus, after the preview time has expired, a total of fourteen minutes of the event may be viewed. An example of how this free time may be used is from 8:30–8:34, 9:10–9:15, and 9:45–9:50. There is no restriction on how the free time is used as long as it does not exceed its predetermined value, in this case, fourteen minutes.

The predetermined value or sum of free time is set by selecting a value for the free time bit pattern F. This value is stored in a non-volatile memory of the IPPV module. Once the subscriber has tuned to a channel on which a IPPV event is being shown, a counter counts down until the free time has elapsed. When the counter equals zero, the event must be purchased to enable further viewing. Thus, each subscriber module has a plurality of counters corresponding to the number of pay channels. A preferred embodiment includes sixteen counters. Fixing the slot bit pattern SL tells the module which of the sixteen counters is to be loaded with the free time. In an alternative embodiment, a single counter of the IPPV module microprocessor is operated by a clock associated with the microprocessor. A memory cell is preferably associated with each channel on which an IPPV event may be shown. The memory cell stores the IPPV event free time which is periodically stepped or decremented responsive to the counter until free time has lapsed for the associated IPPV event channel and, the stepping or decrementing occurs only when the IPPV event on that channel is being displayed and only after preview time has expired. The slot bit pattern SL may be used to tell the module which memory is to be loaded with the free time. It is important to

note that free time is never used up during preview time. During preview time, the free time counter associated with the particular pay channel is constantly reset through receipt of the authorization transaction to the total amount of free time predetermined by the system operator for that event. After preview time, the free time bit pattern F is set to a predetermined value, F(hex) for example, which is a flag to the module that preview time has expired. The free time counter is then allowed to count down or decrement whenever the subscriber is tuned to that particular channel. The counter is decremented on the average every sixty seconds. The actual time is varied slightly in order to defeat attempts at pirating. Both free time and preview time are configurable and represent global commands on a per event basis, which may be downloaded by the system manager.

A security counter controls the length of time that an impulse pay-per-view module will allow the cable TV subscriber to view an impulse pay-per-view channel without receiving an IPPV authorization transaction. When that length of time has elaspsed, the impulse pay-per-view module will deauthorize all impulse pay-per-view channels and "close out" all impulse pay-per-view events that are in progress. This security feature is a method to prevent subscribers from purchasing an event, trapping the data stream, and watching subsequent events.

When an impulse module buys an event, it will authorize that channel until the event is over. In order to effect such channel authorization, the module writes to the set-top's channel authorization memory. When the event is over, a deauthorization transaction is sent to the impulse module to effect channel deauthorization. However, in the past, subscribers have brought a number of premium channels and attemptèd to trap the deauthorization transaction so as to retain use of the premium channels indefinitely. The present invention defeats such attempts at pirating by using the authorization transaction which starts an event and which is sent approximately every five hundred milliseconds. A counter in the module must be reset by one of these authorization transactions within a predetermined period of time, or those channels will be deauthorized. Absence of this transaction will shut down the IPPV channels. The counter is set through the bit pattern values CT0–CT1. The bit values specify the period of the IPPV timer contained in each IPPV module. The timer period is the maximum amount of time that an IPPV module will allow a PPV channel to remain authorized without receiving an authorization transaction from the ATX. The range of valid values in the preferred embodiment is from 0–255 in 15 minute increments. Thus, the time is controllable in fifteen minute increments up to sixty-four hours. The counter may be disabled by setting the bits 0 if there is no concern about piracy.

Another feature of the IPPV system of the present invention is the ability to pre-buy a pay-per-view event. Pre-buys are useful for VCR recording of events when the subscriber is not home or if a subscriber is asleep. This feature of the present invention is unique because of the ability to program the pre-buy either from the IPPV module via a hand-held remote or the set top or from the headend in response to an order via the host billing system.

The process for performing a pre-buy with a Scientific Atlanta Set-top Model 8550 or 8585 is illustrated in FIG. 3. With the converter turned on, the subscriber depresses the keyboard keys "PRG" and "-" of his hand-held remote control. If an access code is required to purchase programming, this must be entered before the converter will enter the IPPV mode and display "VCR" using LED elements. An improperly keyed access code denies one the ability to purchase programming. Once in the IPPV mode, depression of the "AU" key creates access to the pre-buy mode. Once in the pre-buy mode, the subscriber simply enters the three or four digit event ID number of the program he wishes to purchase. The ID numbers may be provided in a programming guide, for example. After the last digit of each program ID has been entered, it is stored in nonvolatile memory **21** of the module. As indicated, it is possible to step through the list of programs which have been pre-bought with an opportunity to cancel any event which the subscriber no longer wishes to view or which have erroneously entered. Although the above description has been given with respect to a particular set-top model, it will be apparent to those of ordinary skill that similar procedures may be employed on different set-tops and the invention should not be understood as limited in this respect.

Subsequently, when the system manager sends an authorization transaction as described above with information that a particular event is running or is authorized, the module, upon receipt of that transaction, will search the nonvolatile memory **21** containing the pre-buys for the event ID of that particular program. If the particular event ID is in fact contained within the nonvolatile memory, the module will purchase the event for the customer. This is an advantage over some prior art systems which require the additional inconvenient setting of times via a hand-held remote control in order to activate the descrambler at a particular time.

In addition, the present system is also adapted to download an IPPV pre-buy in response to a customer's phone request for a PPV event. As schematically indicated in FIG. 1, a converter contains a set-top microprocessor **17** which processes pay-per-view transactions and an impulse pay-per-view microprocessor **22** which processes impulse pay-per-vew transactions. Pay-per-view events are authorized by the set-top microprocessor **17**. If an event is both a pay-per-view event and an impulse pay-per-view event, as frequently happens, and a subscriber buys the event as a pay-per-view event, the microprocessor **22** in the IPPV module would normally cut off viewing after the allocated free time had expired. In order to overcome this, when the system manager is notified that a subscriber wishes to purchase a pay-per-view event and it is determined that this subscriber also has an IPPV module, a transaction as in FIG. **4** is downloaded to the IPPV module which authorizes it to purchase the event. The transaction includes the converter address (bit patterns A0–A4) and the IPPV Event ID (bit patterns I0–I3). This transaction, as in the customer prebuy, loads the Event ID into the module's nonvolatile memory **21**. It is important that the system manager maintains a record of this authorized pre-buy to prevent the record of the impulse purchase from being sent to the billing computer. Such a transfer would result in the subscriber being billed twice for the same event.

In order to report the events watched to the system manager, the IPPV module must be able to dial into a telephone network. Thus, the IPPV module of the present invention possesses the ability to dial a downloaded, stored up to eleven digit telephone number. In place of

a digit, a special character may be used to signify a predetermined delay between the preceding and succeeding digits. This is particularly useful when it is necessary to dial out of a local private branch exchange. A typical load telephone number transaction is included in FIG. 5*a* and 5*b*. This transaction may be global or addressed. FIG. 5*a* includes bit patterns A0–A4 which represent the address of the converter. Bit patterns P0–P10 represent the downloaded telephone number. By inserting a special character, hex A for example, the module may be instructed to pause for a predetermined time period between the preceding and succeeding digits. In the preferred embodiment, the predetermined time period is 2.5 seconds. FIG. 5*b* illustrates a similar global command which does not include address bit patterns A0–A4.

In order to effect a call in, the module must have downloaded to it certain parameters. As noted above, after dialing the phone processor, the phone processor sends a signal to the module indicating it has gone off hook. The module will then send its data. Afterwards the phone processor sends a signal indicating it has received the data. The module must know (a) how long to wait for receipt of the first signal, (b) how many times it should attempt to call back, and (c) when it should start or stop calling back. These parameters may be addressed to a particular set-top or globally.

These parameters may be sent as part of the transaction shown in FIG. 6. The bit patterns TL0–TL1 represent the call back attempt limit and may include values from zero to FF, with zero used to instruct the module to stop calling. L0–L1 represent the host time out or how long the module will wait after dialing the last digit for the first signal from the phone processor. If no response is received, the IPPV module will consider the call to be unsuccessful and retry at a later time. The range of valid values is zero to 255, with a resolution of 2.5 seconds.

The impulse pay-per-view system also has the ability to request the impulse pay-per-view module to record the channel that is being watched and report that information during the normal IPPV call back. If the cable operator wishes to find out what his subscribers are watching, a transaction may be sent similar to that shown in FIG. 7. All set tops with an IPPV module will record the channel being watched upon receipt of this transaction. After the module has called in to report its viewing statistics, a third bit is stored in the nonvolatile memory to indicate that the statistics have already been reported. A value of '00' may be used to indicate that a set-top was off when the Viewing Statistic Transaction was transmitted and a value of 'FF' may be used to indicate that a set-top has already transferred its channel information. Thus, if that particular IPPV module calls in twice before the cable operator has had a chance to calculate the statistics, the statistics of that particular IPPV module are not duplicated.

It is to be understood that the invention is not limited to the illustrations described and shown herein which are deemed to be merely illustrative of the best modes of carrying out the invention. The invention rather is intended to encompass all such modifications which are within its spirit and scope as defined by the claims.

We claim:

1. A subscriber terminal apparatus for controlling subscriber viewing in an addressable television system, the subscriber terminal apparatus comprising:

receiving means for receiving a television signal comprising a plurality of channels, at least one of said channels carrying pay-per-view events which must be authorized for viewing by a subscriber;

detecting means for detecting data transactions from a headend which are addressed to the subscriber terminal apparatus;

microprocessor means for processing the data transactions detected by said detecting means;

authorization means responsive to subscriber-supplied signals for authorizing channels carrying pay-per-view events and for generating billing information related to said authorized pay-per-view events;

storage means coupled to said microprocessor means and said authorization means for storing the billing information;

transmitting means coupled to said storage means for transmitting the stored billing information to said headend;

preview time means responsive to a first data transaction of said data transactions which are addressed to the subscriber terminal apparatus for permitting the subscriber to view a selected pay-per-view event for a predetermined preview time period before authoization of said selected pay-per-view event; and

free time means responsive to a second data transaction of said data transactions which are addressed to the subscriber terminal apparatus for further permitting the subscriber to view said selected pay-per-view event for a predetermined free time period before authorization of said selected pay-per-view event.

2. The subscriber terminal apparatus according to claim 1 wherein said free time means comprises:

a plurality of counters, a single counter associated with each channel carrying pay-per-view events.

3. The subscriber terminal apparatus according to claim 2 wherein said plurality of counters comprises sixteen counters.

4. The subscriber terminal apparatus according to claim 2 wherein each of said plurality of counters may be set to a maximum of fourteen minutes.

5. The subscriber terminal apparatus according to claim 2 wherein the counter associated with a particular channel carrying a pay-per-view event is continually to a predetermined free time period during the preview time period.

6. The subscriber terminal apparatus according to claim 2 wherein the counter associated with a particular channel carrying a pay-per-view event is decremented when the subscriber selects the channel carrying said particular pay-per-view event.

7. The subscriber terminal apparatus according to claim 6 wherein each counter is decremented approximately every 60 seconds.

8. The subscriber terminal apparatus according to claim 1 wherein said predetermined free time period may be selectively allocated.

9. The subscriber terminal apparatus according to claim 1 wherein said preview time may range from 0 to 255 minutes.

10. The subscriber terminal apparatus according to claim 2 wherein said microprocessor means sets the respective counters in accordance with slot information contained in said first data transaction.

11. The subscriber terminal apparatus according to claim 1, the free time means further comprising memory means for storing the predetermined free time period.

12. The subscriber terminal apparatus according to claim 1, the free time means further comprising a counter for counting time lapsed of the predetermined free time period.

13. The subscriber terminal apparatus according to claim 1, the microprocessor means further processing said billing information and controlling pay-per-view event status.

14. The subscriber terminal apparatus according to claim 13, wherein the microprocessor means comprises a first microprocessor for processing the detected transactions and for controlling the pay-per-view event status and a second microprocessor, responsive to the first microprocessor, for processing the billing information.

15. The subscriber terminal apparatus according to claim 11, the free time means further comprising a counter for counting time lapsed of the predetermined free time period.

16. The subscriber terminal apparatus according to claim 15, the memory means of the free time means comprising a plurality of memories, a single memory being associated with each channel carrying pay-per-view events.

17. The subscriber terminal apparatus according to claim 1 wherein said data transactions are contained in said television signal.

18. The subscriber terminal apparatus according to claim 1 wherein said data transactions are transmitted on a dedicated data channel.

19. The subscriber terminal apparatus according to claim 1 wherein said transmitting means is responsive to a third data transaction to effect dialing into said telephone network.

20. The subscriber terminal apparatus according to claim 19 wherein said third data transaction includes a telephone number for dialing a storage means for storing billing information associated with a plurality of subscribers.

21. The subscriber terminal apparatus according to claim 20 wherein said third data transaction includes data for effecting predetermined pauses in the dialing of said telephone number.

22. The subscriber terminal according apparatus to claim 1 wherein said addressable television system comprises an addressable cable television system.

23. The subscriber terminal apparatus according to claim 1 wherein said transmitting means transmits the stored billing information to said headend office over a telephone network.

24. Headend apparatus for an addressable television system, the headend apparatus comprising:

transmitting means for transmitting a television signal and a plurality of data transactions to a subscriber terminal apparatus, said television signal comprising a plurality of channels, at least one of said channels carrying pay-per-view events having respective predetermined active periods which may be selectively authorized for viewing by a subscriber so as to generate billing information;

preview time control means for generating a first data transaction of said plurality of data transactions which are transmitted to the subscriber terminal apparatus for permitting the subscriber terminal apparatus to receive a subscriber selected pay-per-view-event for a predetermined preview time period before authorization of said selected pay-per-view event; and

free time control means for generating a second data transaction of said plurality of data transactions which are transmitted to the subscriber terminal apparatus for permitting the subscriber terminal apparatus to further receive said subscriber selected pay-per-view event for a predetermined free time period before authorization of said selected pay-per-view event.

25. The headend apparatus apparatus according to claim 24 wherein said data transactions are contained in said television signal.

26. The headend apparatus according to claim 24 wherein said data transactions are transmitted on a dedicated data channel.

27. The headend apparatus according to claim 24 further comprising:

storage means for storing billing information generated by a plurality of subscribers.

28. The headend apparatus according to claim 27 wherein said storage means is adapted to receive billing information over a telephone network.

29. The headend apparatus according to claim 24 further comprising:

dial control means for generating a third data transaction for instructing the subscriber terminal to dial into a telephone network.

30. The headend apparatus according to claim 29 wherein said third data transaction includes a telephone number for dialing a storage means for storing billing information associated with a plurality of subscribers.

31. The headend apparatus according to claim 30 wherein said third data transaction includes data for effecting predetermined pauses in the dialing of said telephone number.

32. The headend apparatus according to claim 24 further comprising:

deauthorization control means for generating a fourth data transaction transmitted only during the predetermined active period of pay-per-view event for deauthorizing an authorized channel after a predetermined period of time.

33. A subscriber terminal apparatus for controlling subscriber viewing in an addressable television system, the subscriber terminal apparatus comprising:

receiving means for receiving a television signal comprising a plurality of channels, at least one of said channels carrying pay-per-view events which must be authorized for viewing by a subscriber;

detecting means for detecting data transactions from a headend which are addressed to the subscriber terminal apparatus;

microprocessor means for processing the data transactions detected by said detecting means;

authorization means responsive to subscriber-supplied signals for authorizing channels carrying pay-per-view events having respective predetermined active times and for generating billing information related to said authorized pay-per-view events;

storage means coupled to said microporcessor means and said authorization means for storing said billing information;

transmitting means coupled to said storage means for transmitting the stored billing information to the headend; and

deauthorization means responsive to a first data transaction of the data transactions which are addressed to the subscriber terminal apparatus which is sent only during the predetermined active time of a pay-per-view event for deauthorizing an authorized channel after a predetermined period of time.

34. The subscriber terminal apparatus according to claim **33** wherein said deauthorization means comprises:

a security counter set in accordance with instructions contained in said first data transaction.

35. The subscriber terminal apparatus according to claim **34** wherein said security counter decrements in fifteen minute intervals.

36. The subscriber terminal apparatus according to claim **33** wherein said first data transaction is sent at predetermined times during the active time of a selected event.

37. The subscriber terminal apparatus according to claim **33** wherein said first data transaction is sent at regular intervals during the active time of a selected event.

38. The subscriber terminal apparatus according to claim **37** wherein said first data transation is sent approximately every 500 milliseconds during the active time of a selected event.

39. The subscriber terminal apparatus according to claim **34** wherein said first data transaction is sent at predetermined times during the active time of a selected event.

40. The subscriber terminal apparatus according to claim **34** wherein said security counter is reset in response to said first data transaction.

41. The subscriber terminal apparatus according to claim **33** wherein said security means may be selectively disabled.

42. Headend apparatus for an addressable teleision system, the headend apparatus comprising:

transmitting means for transmitting a television signal and a plurality of data transactions to a subscriber terminal apparatus, said television signal comprising a plurality of channels, at least one of said channels carrying pay-per-view events having respective predetermined active periods which may be selectively authorized for viewing by a subscriber;

deauthorization control means for generating and for transmitting a first data transaction of said plurality of data transactions which are transmitted to the subscriber terminal apparatus only during the predetermined active period of a pay-per-view event for deauthorizing an authorized channel after a predetermined period of time.

43. A method of generating viewer statistics in an addressable television system which distributes a television signal from a headend to a subscriber terminal apparatus including a storage device, said television signal comprising a plurality of channels, the method comprising the steps of:

transmitting a first data transaction to the subscriber terminal apparatus, said first data transaction including instructions to store in the storage device the channel number corresponding to the channel tuned by the subscriber terminal apparatus when said first data transaction is received;

transmitting a second data transaction to the subscriber terminal apparatus, said second data transaction including instructions to the subscriber terminal apparatus to initiate telephone communication with said headend, said instructions further including instructions for transferring the stored channel number to said headend and for writing a predetermined character to said storage device as an indication that the channel number has been transferred.

44. The method according to claim **43** wherein said first and said second data transactions are contained in said television signal.

45. The method according to claim **43** wherein said first and said second data transactions are transmitted over a dedicated data channel.

46. A method of generating statistics in a cable television system for distributing a television signal from a headend to a subscriber terminal apparatus including a storage device, said television signal comprising a plurality of channels, the method of comprising the steps of:

storing a channel number corresponding to the channel tuned by the subscriber terminal apparatus in the storage device in response to a first data transaction from said headend;

dialing into the public switched telephone network in response to a second data transaction from said headend to establish communication with said headend;

transferring said stored channel number to said headend; and

writing a predetermined character to said storage device as an indication that the stored channel number has been transferred.

47. The method according to claim **46** wherein said first data transaction is contained in said television signal.

48. The method according to claim **46** wherein said first data transaction is received over a dedicated data channel.

* * * * *

US006317885B1

(12) **United States Patent**
Fries

(10) **Patent No.: US 6,317,885 B1**
(45) **Date of Patent: Nov. 13, 2001**

(54) **INTERACTIVE ENTERTAINMENT AND INFORMATION SYSTEM USING TELEVISION SET-TOP BOX**

(75) Inventor: **Robert M. Fries**, Redmond, WA (US)

(73) Assignee: **Microsoft Corporation**, Redmond, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **08/882,908**

(22) Filed: **Jun. 26, 1997**

(51) **Int. Cl.**[7] **H04N 7/173**
(52) **U.S. Cl.** **725/109**; 725/37
(58) **Field of Search** 345/327, 302; 709/219; 348/10, 12, 13, 7; 455/5.1, 6.2, 6.3, 4.2, 3.1; 725/109, 110, 111, 112, 113, 37; 707/501, 543

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,694,163 * 12/1997 Harrison 348/13
5,903,816 * 5/1999 Broadwin et al. 455/5.1
5,929,850 * 7/1999 Broadwin et al. 345/327
5,982,445 * 11/1999 Eyer et al. 348/461
6,018,764 * 1/2000 Field et al. 709/217

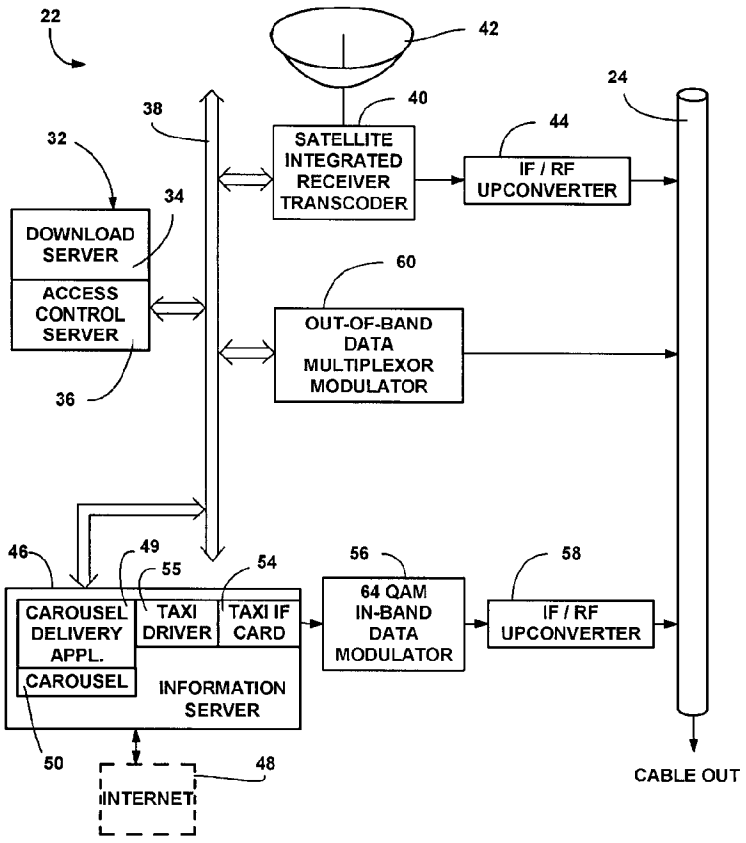
* cited by examiner

Primary Examiner—Chris Grant
(74) *Attorney, Agent, or Firm*—Michalik & Wylie, PLLC

(57) **ABSTRACT**

An interactive entertainment and information system using a television set-top box, wherein pages of information are periodically provided to the set-top box for user interaction therewith. The pages include associated meta-data defining active locations on each page. When a page is displayed, the user interacts with the active locations on the page by entering commands via a remote control device, whereby the system reads the meta-data and takes the action associated with the location. Actions include moving to other active locations, hyperlinking to other pages, entering user form data and submitting the data as a form into memory. The form data may be read from memory, and the pages may be related to a conventional television program, thereby providing significant user interactivity with the television.

34 Claims, 13 Drawing Sheets



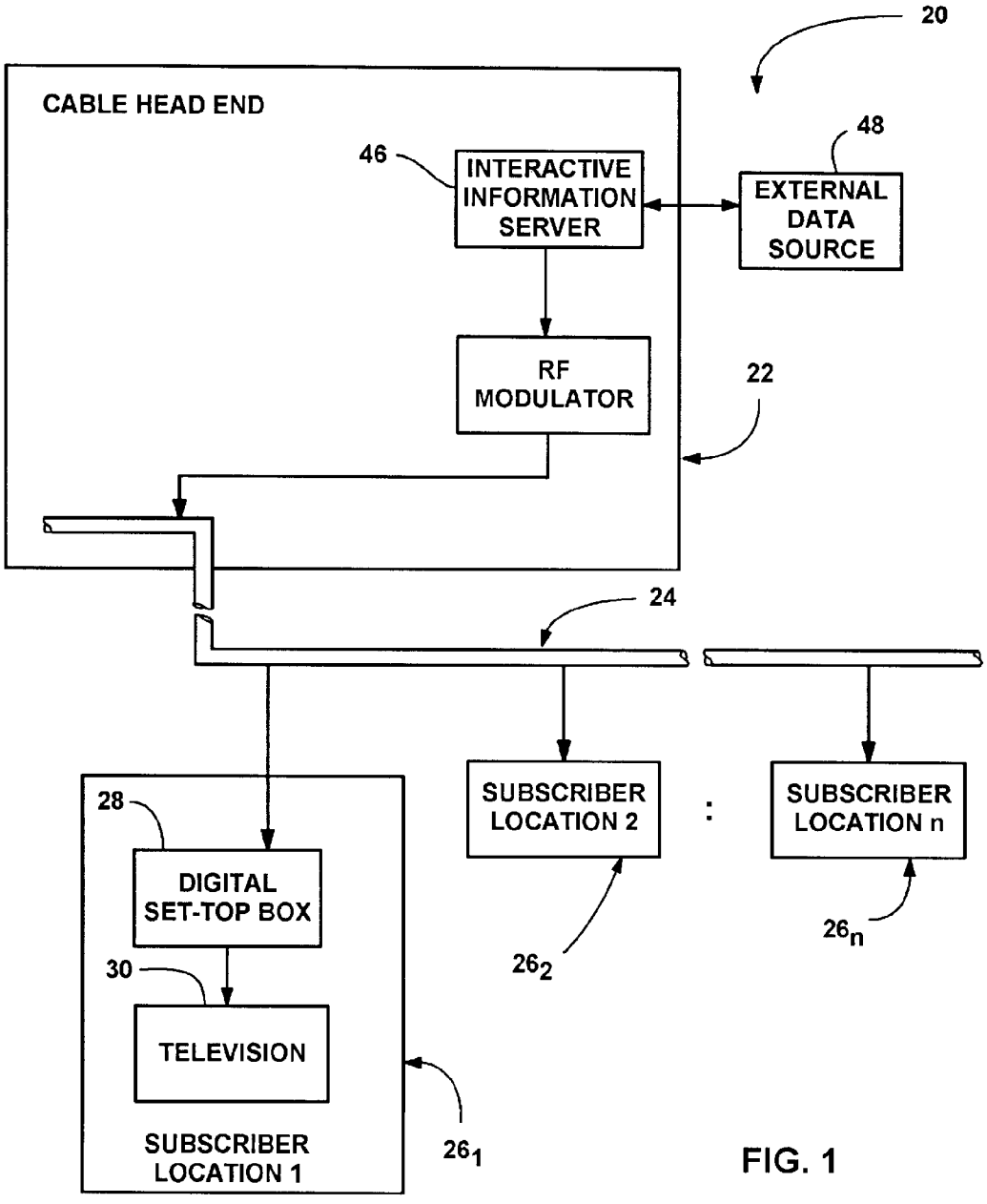

20
CABLE HEAD END
46
INTERACTIVE INFORMATION SERVER
48
EXTERNAL DATA SOURCE
RF MODULATOR
22
24
28
DIGITAL SET-TOP BOX
30
TELEVISION
SUBSCRIBER LOCATION 1
26_1
SUBSCRIBER LOCATION 2
26_2
SUBSCRIBER LOCATION n
26_n

FIG. 1

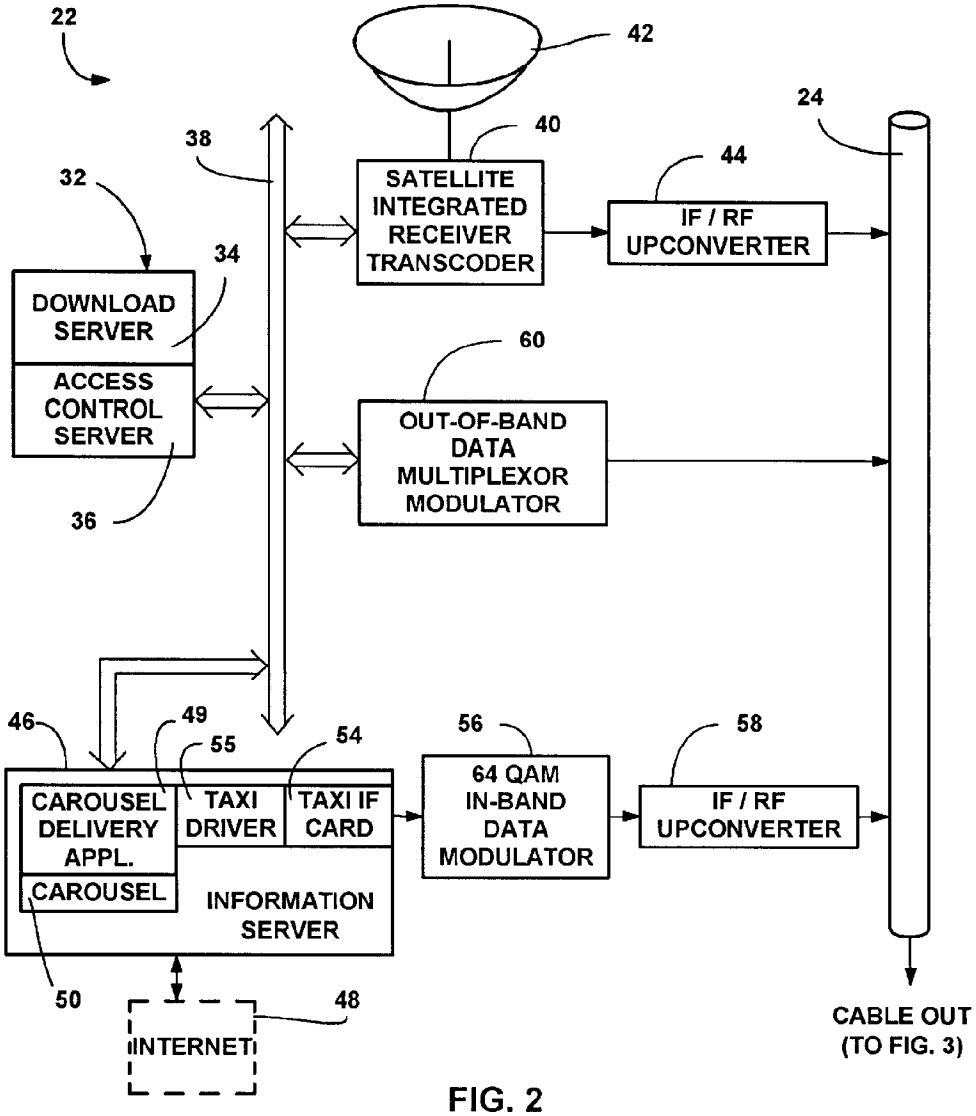

22
42
24
40
38
44
32
SATELLITE INTEGRATED RECEIVER TRANSCODER
IF / RF UPCONVERTER
34
DOWNLOAD SERVER
ACCESS CONTROL SERVER
60
OUT-OF-BAND DATA MULTIPLEXOR MODULATOR
36
46
49
55
54
56
58
CAROUSEL DELIVERY APPL.
TAXI DRIVER
TAXI IF CARD
64 QAM IN-BAND DATA MODULATOR
IF / RF UPCONVERTER
CAROUSEL
INFORMATION SERVER
50
48
INTERNET
CABLE OUT
(TO FIG. 3)

FIG. 2

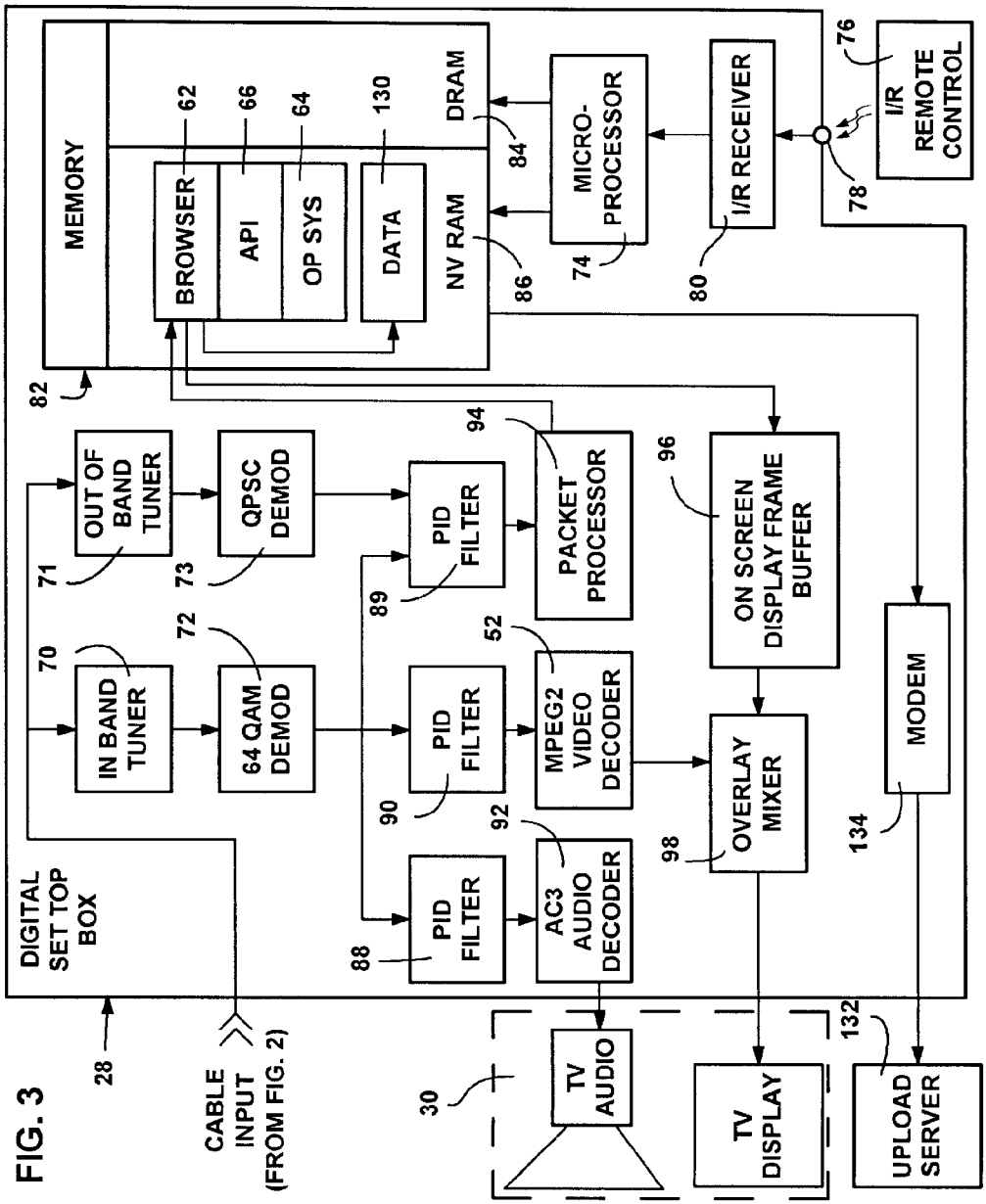

DIGITAL SET TOP BOX
28
CABLE INPUT (FROM FIG. 2)
IN BAND TUNER
70
OUT OF BAND TUNER
71
64 QAM DEMOD
72
QPSC DEMOD
73
PID FILTER
90
PID FILTER
89
PID FILTER
88
MPEG2 VIDEO DECODER
52
AC3 AUDIO DECODER
92
PACKET PROCESSOR
94
MEMORY
BROWSER
62
API
66
OP SYS
64
DATA
130
DRAM
NV RAM
86
MICRO-PROCESSOR
74
84
I/R RECEIVER
80
I/R REMOTE CONTROL
76
78
ON SCREEN DISPLAY FRAME BUFFER
96
OVERLAY MIXER
98
MODEM
134
TV AUDIO
30
TV DISPLAY
UPLOAD SERVER
132

FIG. 3

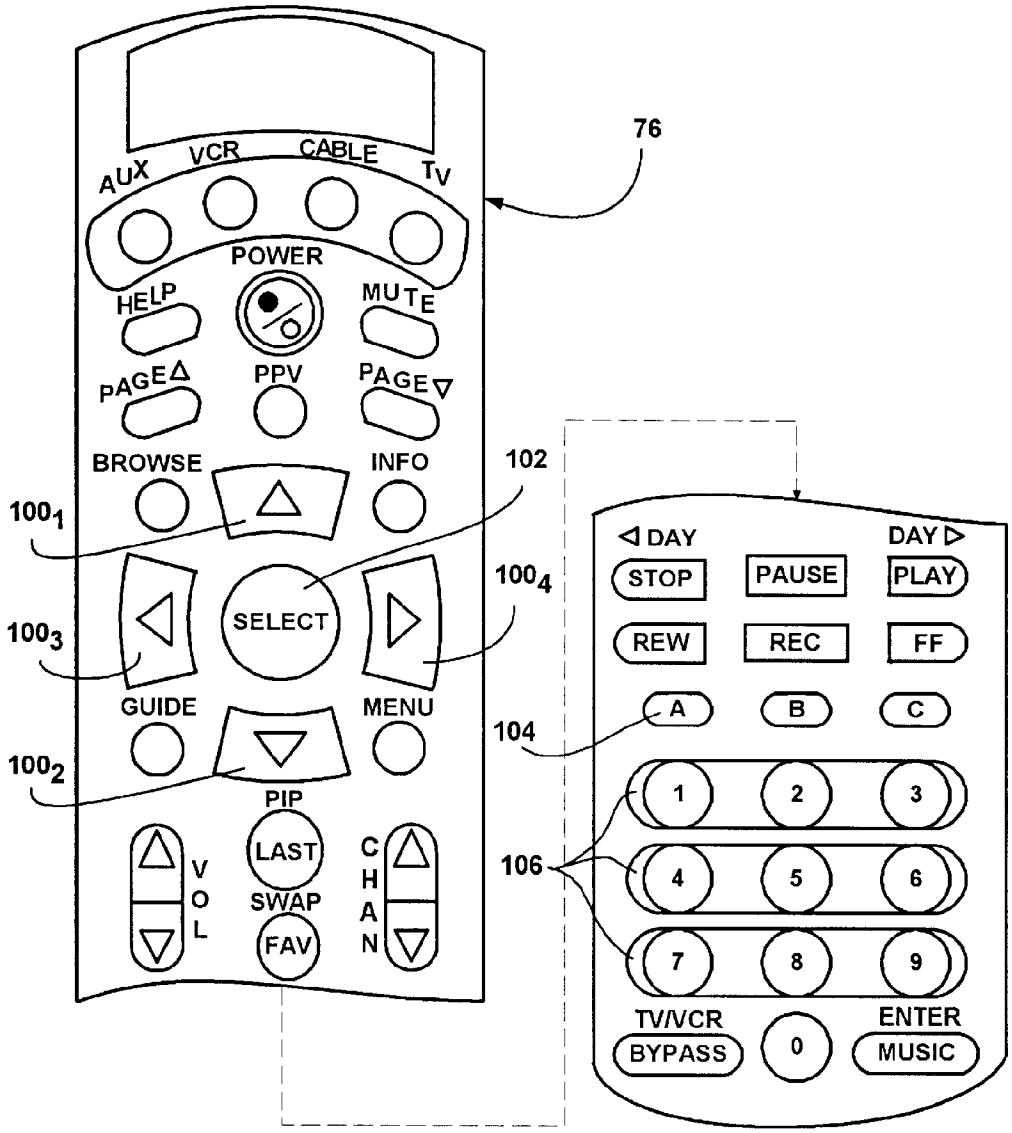

76
AUX
VCR
CABLE
TV
POWER
HELP
MUTE
PAGE
PPV
PAGE
BROWSE
INFO
102
100₁
100₄
100₃
SELECT
GUIDE
MENU
104
100₂
PIP
LAST
SWAP
FAV
VOL
CHAN
106
DAY
DAY
STOP
PAUSE
PLAY
REW
REC
FF
A
B
C
1
2
3
4
5
6
7
8
9
TV/VCR
BYPASS
0
ENTER
MUSIC

FIG. 4

FIG. 5A

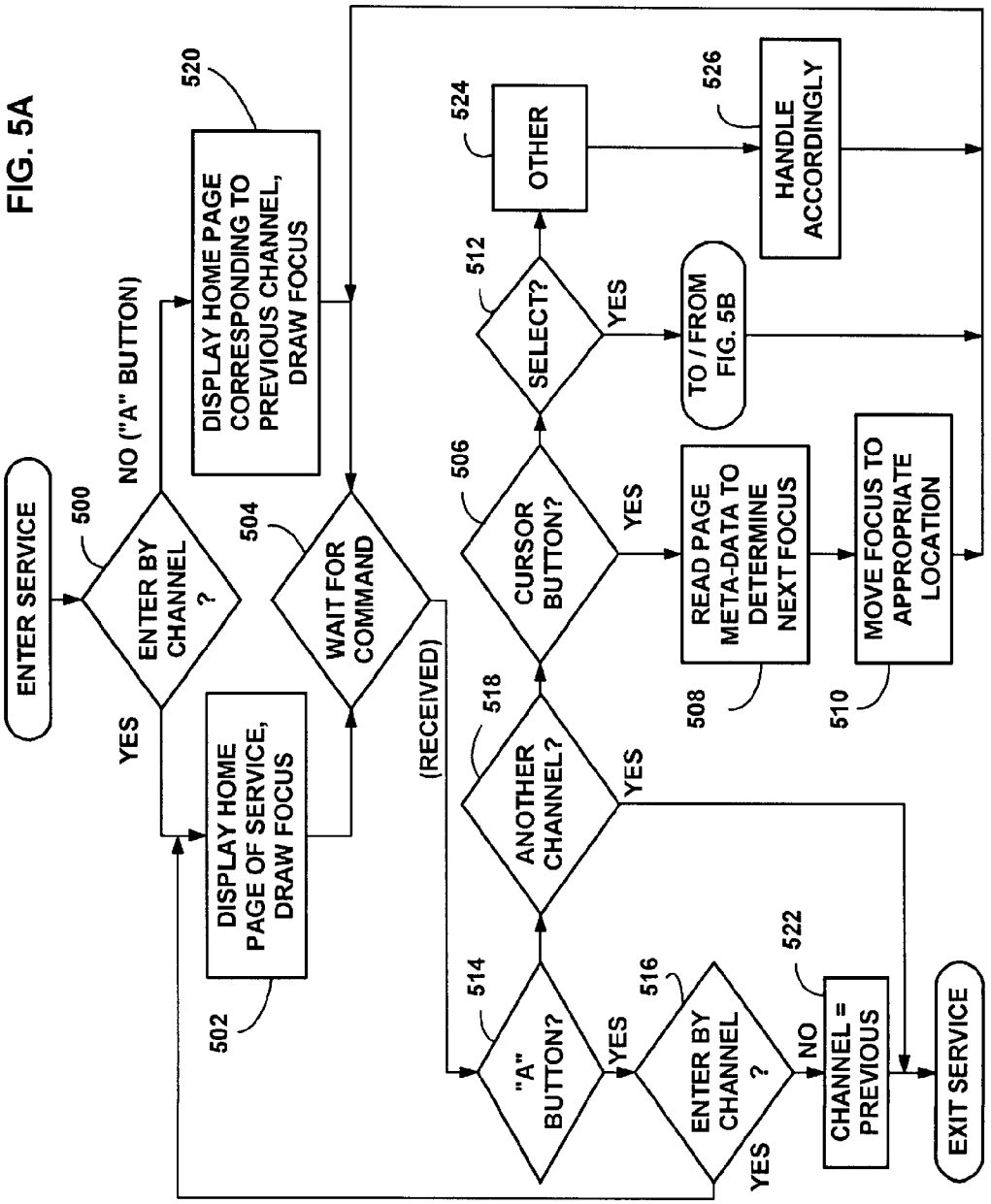
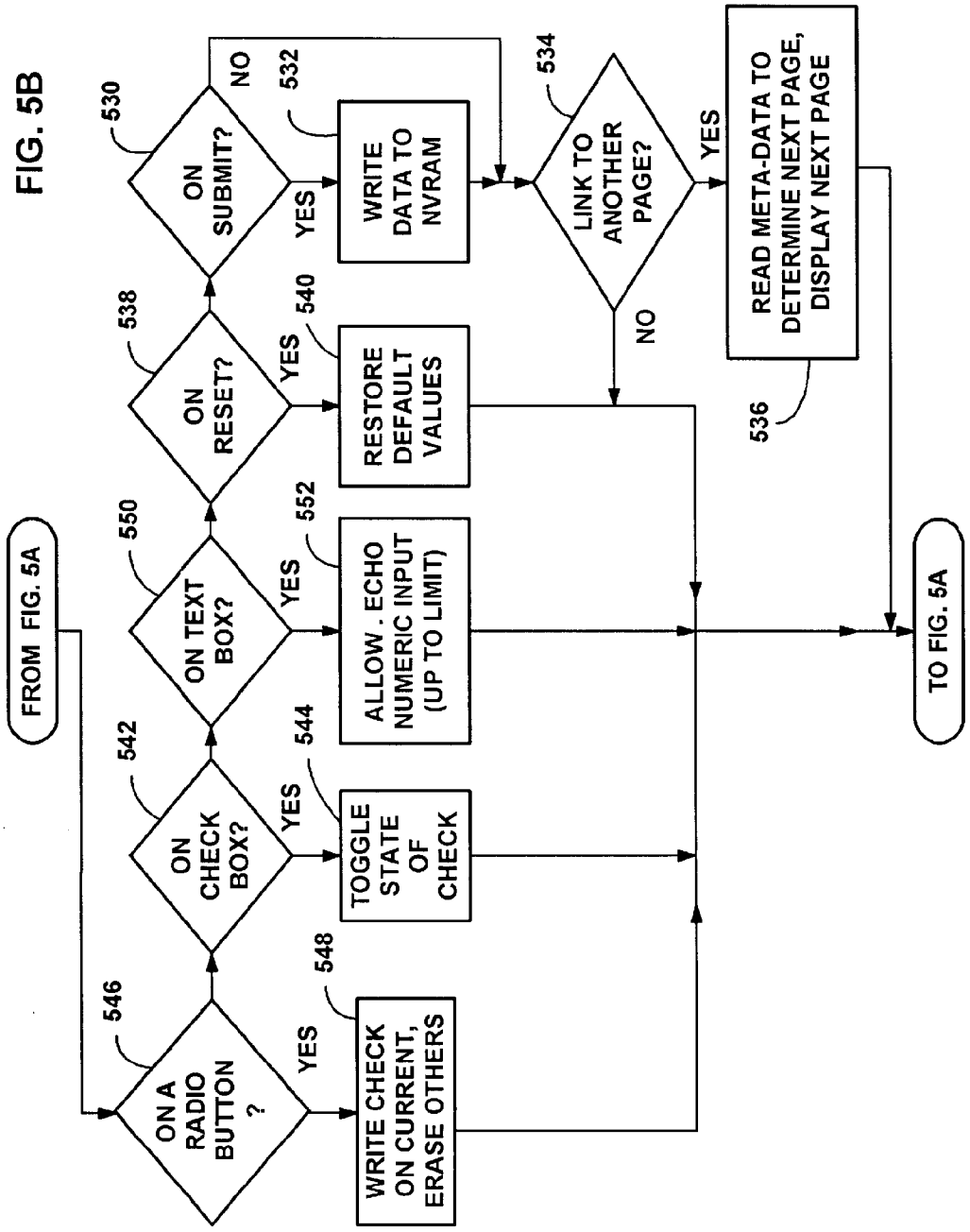

ENTER SERVICE
500
ENTER BY CHANNEL ?
NO ("A" BUTTON)
YES
DISPLAY HOME PAGE CORRESPONDING TO PREVIOUS CHANNEL, DRAW FOCUS
520
DISPLAY HOME PAGE OF SERVICE, DRAW FOCUS
502
504
WAIT FOR COMMAND
(RECEIVED)
506
CURSOR BUTTON?
YES
READ PAGE META-DATA TO DETERMINE NEXT FOCUS
508
MOVE FOCUS TO APPROPRIATE LOCATION
510
512
SELECT?
YES
TO / FROM FIG. 5B
524
OTHER
526
HANDLE ACCORDINGLY
518
ANOTHER CHANNEL?
YES
514
"A" BUTTON?
YES
516
ENTER BY CHANNEL ?
NO
YES
522
CHANNEL = PREVIOUS
EXIT SERVICE

FIG. 5B


FROM FIG. 5A
ON A RADIO BUTTON ?
546
YES
WRITE CHECK ON CURRENT, ERASE OTHERS
548
ON CHECK BOX?
542
YES
TOGGLE STATE OF CHECK
544
ON TEXT BOX?
550
YES
ALLOW . ECHO NUMERIC INPUT (UP TO LIMIT)
552
ON RESET?
538
YES
RESTORE DEFAULT VALUES
540
ON SUBMIT?
530
NO
YES
WRITE DATA TO NVRAM
532
LINK TO ANOTHER PAGE?
534
YES
NO
READ META-DATA TO DETERMINE NEXT PAGE, DISPLAY NEXT PAGE
536
TO FIG. 5A

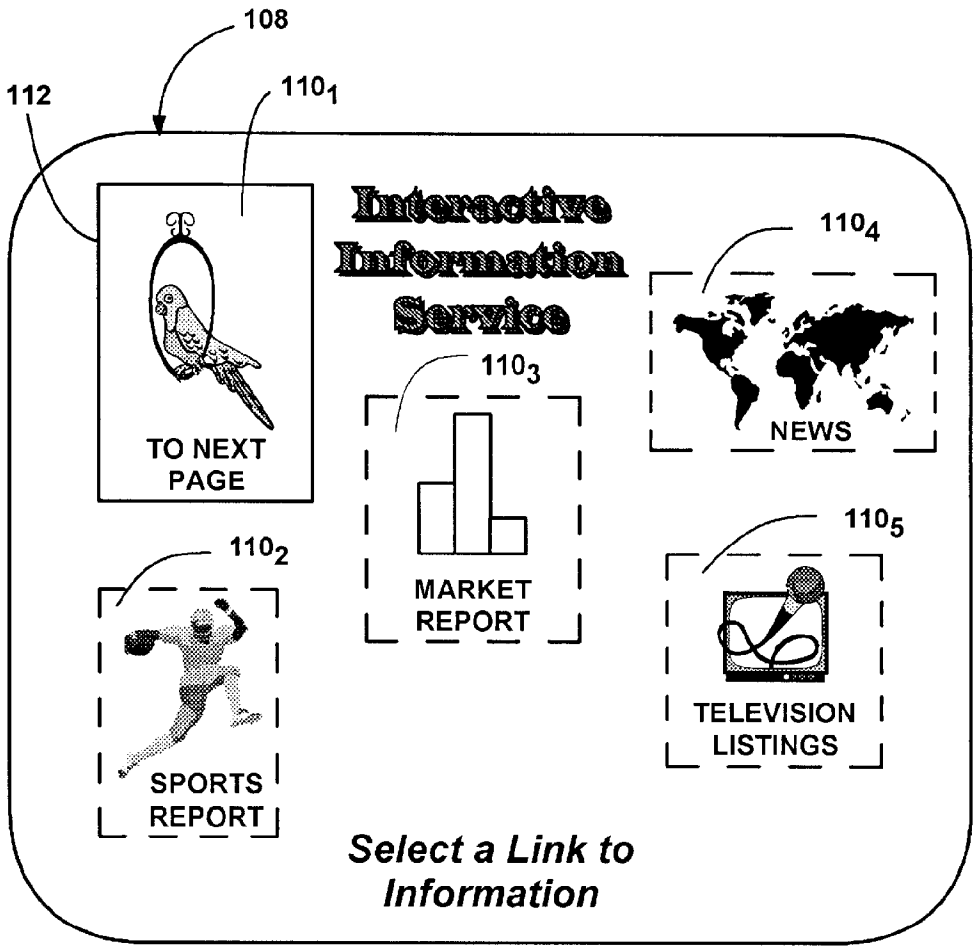
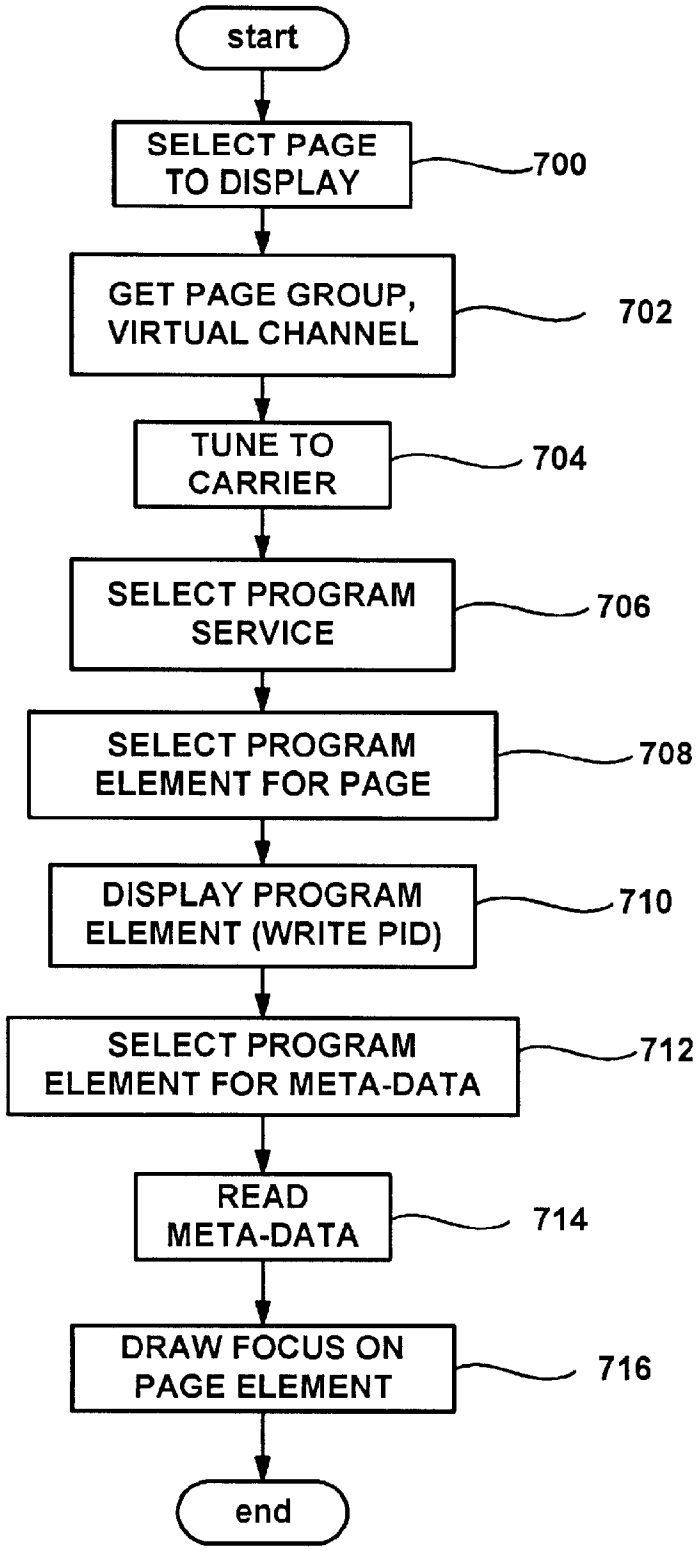

108
112
110_1
Interactive Information Service
TO NEXT PAGE
110_2
SPORTS REPORT
110_3
MARKET REPORT
110_4
NEWS
110_5
TELEVISION LISTINGS
Select a Link to Information

FIG. 6

FIG. 7


start
SELECT PAGE TO DISPLAY
700
GET PAGE GROUP, VIRTUAL CHANNEL
702
TUNE TO CARRIER
704
SELECT PROGRAM SERVICE
706
SELECT PROGRAM ELEMENT FOR PAGE
708
DISPLAY PROGRAM ELEMENT (WRITE PID)
710
SELECT PROGRAM ELEMENT FOR META-DATA
712
READ META-DATA
714
DRAW FOCUS ON PAGE ELEMENT
716
end

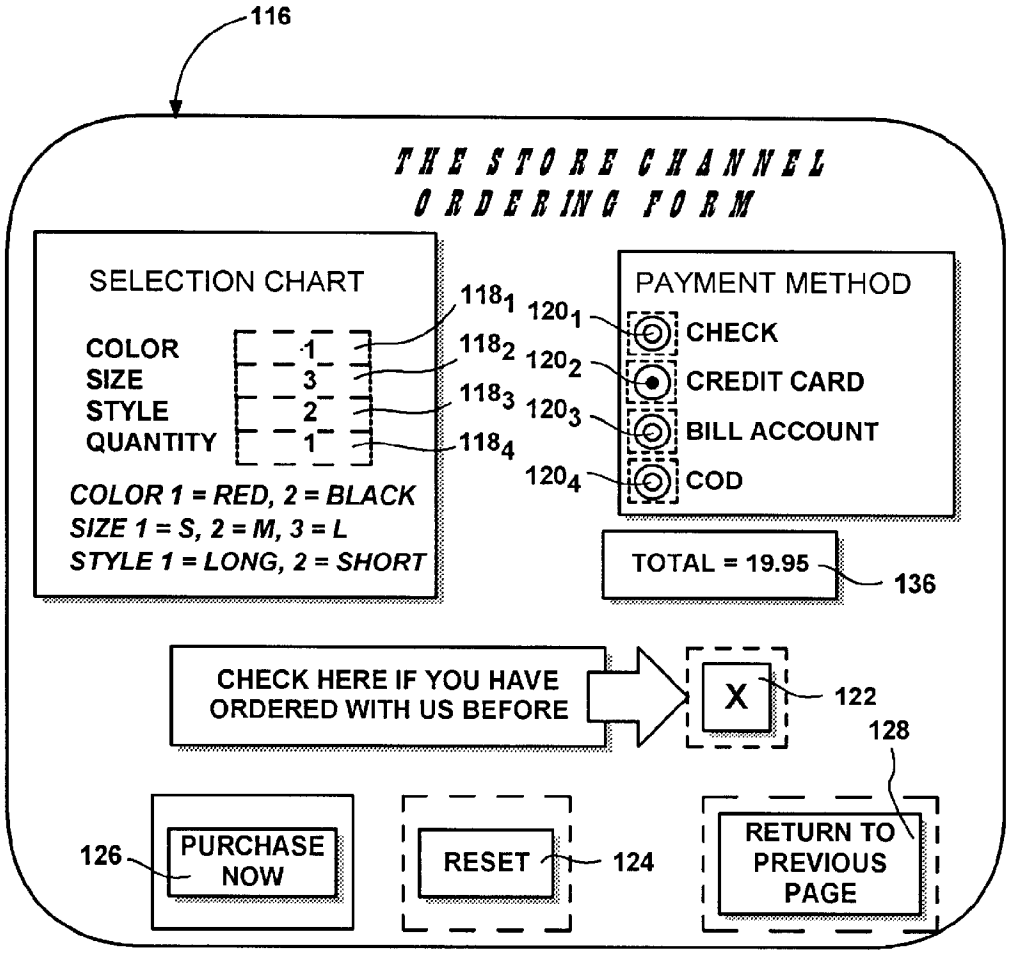

116
THE STORE CHANNEL
ORDERING FORM
SELECTION CHART
COLOR 1
SIZE 3
STYLE 2
QUANTITY 1
COLOR 1 = RED, 2 = BLACK
SIZE 1 = S, 2 = M, 3 = L
STYLE 1 = LONG, 2 = SHORT
118₁
118₂
118₃
118₄
PAYMENT METHOD
120₁ CHECK
120₂ CREDIT CARD
120₃ BILL ACCOUNT
120₄ COD
TOTAL = 19.95
136
CHECK HERE IF YOU HAVE ORDERED WITH US BEFORE
X
122
128
126
PURCHASE NOW
RESET
124
RETURN TO PREVIOUS PAGE

FIG. 8

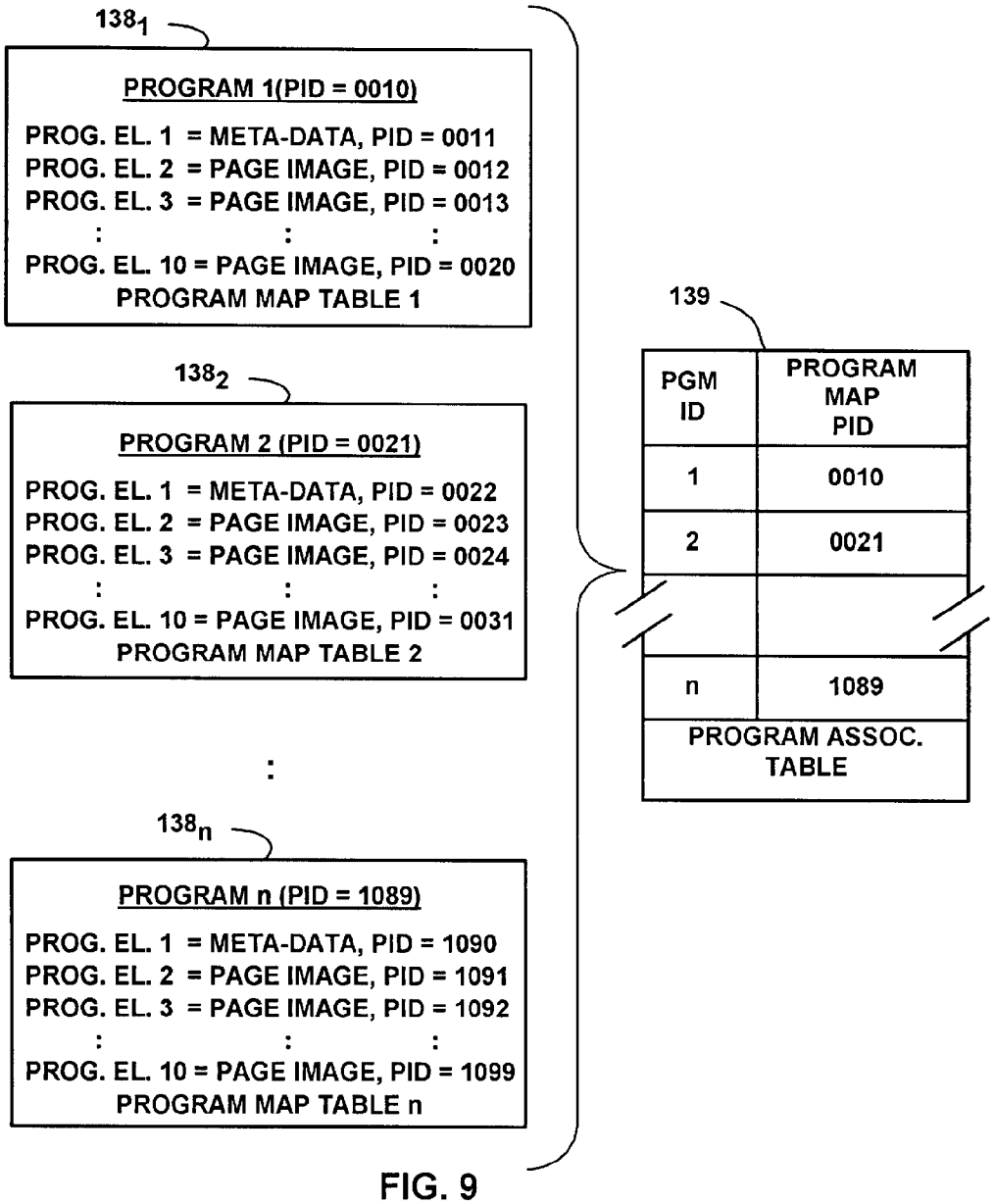

138_1
PROGRAM 1(PID = 0010)
PROG. EL. 1 = META-DATA, PID = 0011
PROG. EL. 2 = PAGE IMAGE, PID = 0012
PROG. EL. 3 = PAGE IMAGE, PID = 0013
PROG. EL. 10 = PAGE IMAGE, PID = 0020
PROGRAM MAP TABLE 1
138_2
PROGRAM 2 (PID = 0021)
PROG. EL. 1 = META-DATA, PID = 0022
PROG. EL. 2 = PAGE IMAGE, PID = 0023
PROG. EL. 3 = PAGE IMAGE, PID = 0024
PROG. EL. 10 = PAGE IMAGE, PID = 0031
PROGRAM MAP TABLE 2
138_n
PROGRAM n (PID = 1089)
PROG. EL. 1 = META-DATA, PID = 1090
PROG. EL. 2 = PAGE IMAGE, PID = 1091
PROG. EL. 3 = PAGE IMAGE, PID = 1092
PROG. EL. 10 = PAGE IMAGE, PID = 1099
PROGRAM MAP TABLE n
139
PGM ID
PROGRAM MAP PID
1
0010
2
0021
n
1089
PROGRAM ASSOC. TABLE

FIG. 9

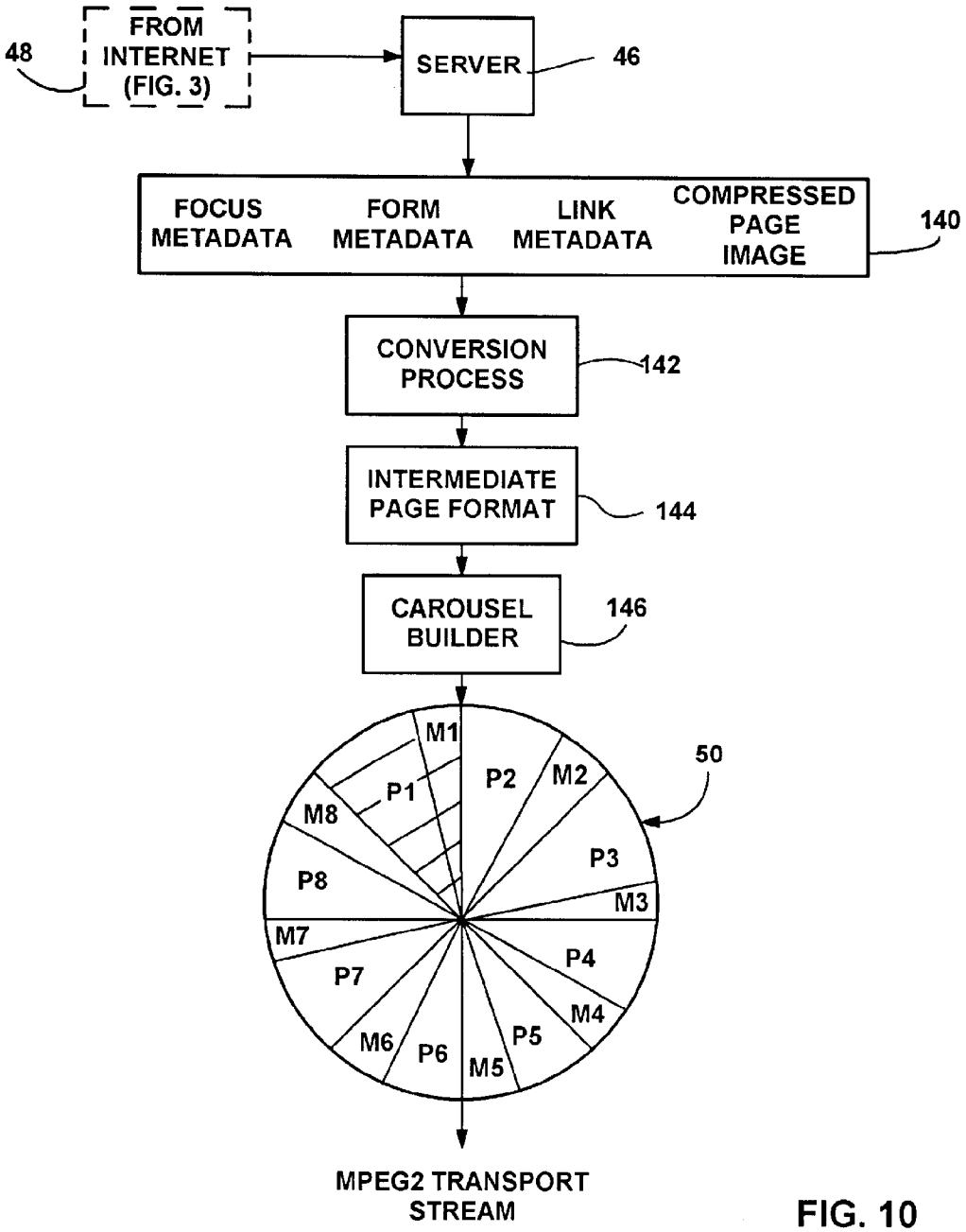
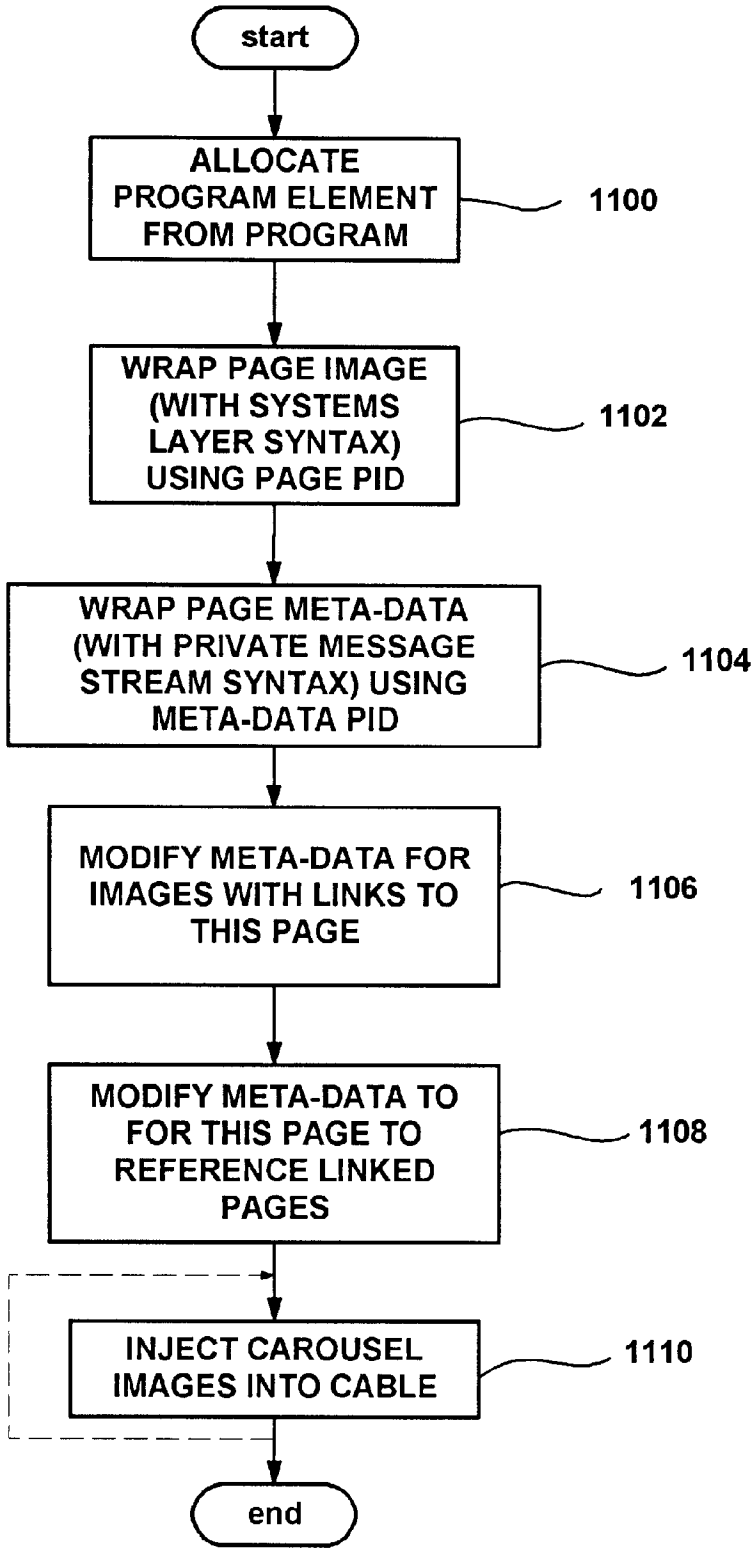

48
FROM INTERNET (FIG. 3)
SERVER
46
FOCUS METADATA
FORM METADATA
LINK METADATA
COMPRESSED PAGE IMAGE
140
CONVERSION PROCESS
142
INTERMEDIATE PAGE FORMAT
144
CAROUSEL BUILDER
146
50
M1
P1
P2
M2
M8
P3
P8
M3
M7
P4
P7
M4
M6
P6
P5
M5
MPEG2 TRANSPORT STREAM

FIG. 10

FIG. 11


start
ALLOCATE PROGRAM ELEMENT FROM PROGRAM
1100
WRAP PAGE IMAGE (WITH SYSTEMS LAYER SYNTAX) USING PAGE PID
1102
WRAP PAGE META-DATA (WITH PRIVATE MESSAGE STREAM SYNTAX) USING META-DATA PID
1104
MODIFY META-DATA FOR IMAGES WITH LINKS TO THIS PAGE
1106
MODIFY META-DATA TO FOR THIS PAGE TO REFERENCE LINKED PAGES
1108
INJECT CAROUSEL IMAGES INTO CABLE
1110
end

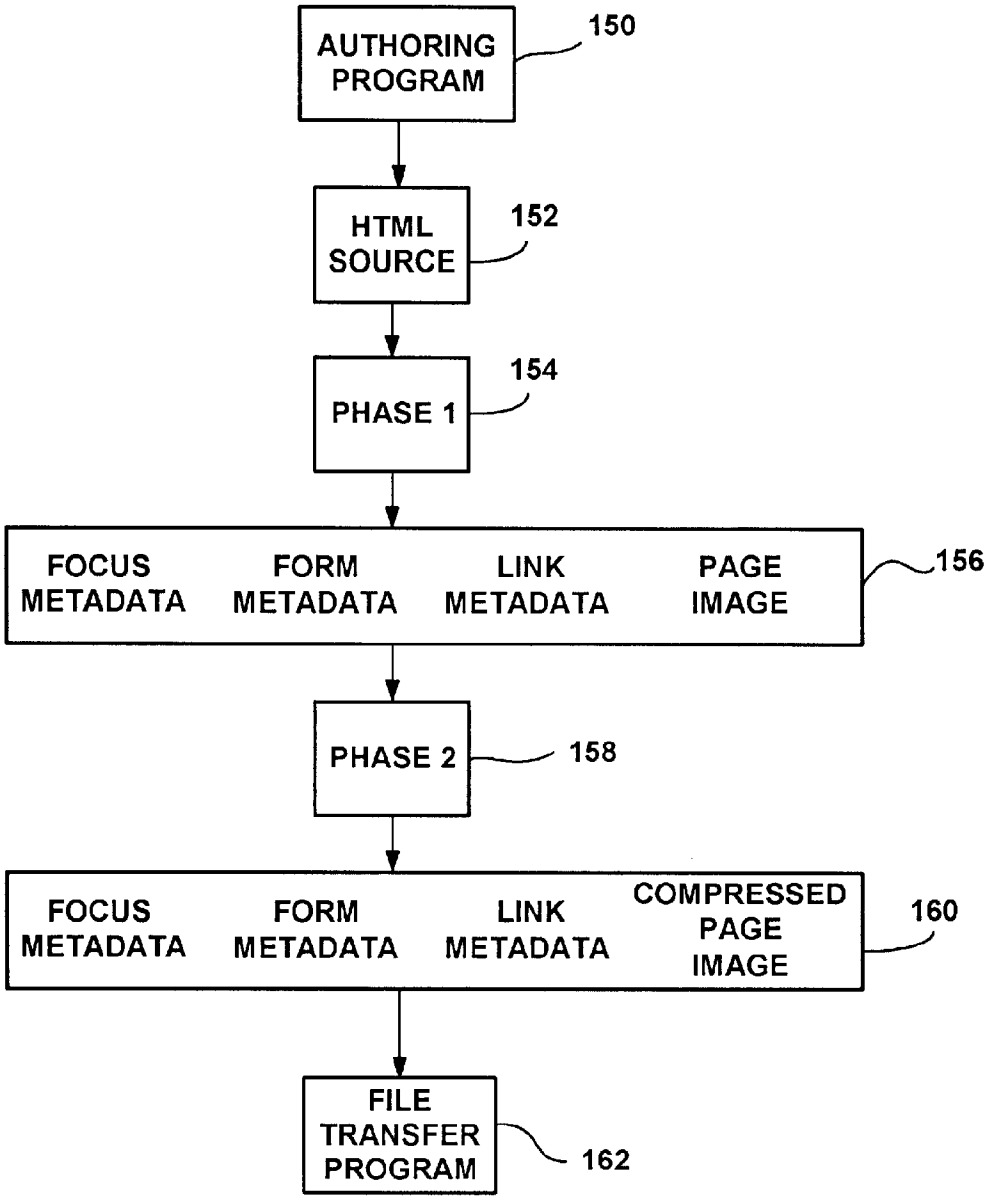

AUTHORING PROGRAM
150
HTML SOURCE
152
PHASE 1
154
FOCUS METADATA
FORM METADATA
LINK METADATA
PAGE IMAGE
156
PHASE 2
158
FOCUS METADATA
FORM METADATA
LINK METADATA
COMPRESSED PAGE IMAGE
160
FILE TRANSFER PROGRAM
162

FIG. 12

INTERACTIVE ENTERTAINMENT AND INFORMATION SYSTEM USING TELEVISION SET-TOP BOX

FIELD OF THE INVENTION

The invention relates generally to interactive television, and more particularly to an interactive entertainment and information system using a cable television set-top box.

BACKGROUND OF THE INVENTION

The existing network of coaxial cable television wiring was once foreseen as providing the primary high-speed infrastructure for linking computers and computerized televisions to sources of information such as the Internet. However, it is becoming apparent that the availability of such cable-based information access will not be realized in the near term, if ever, particularly as competing high-speed transmission media (such as that based on higher quality telephone lines, faster modems and ISDN technology) become entrenched. Indeed, the current economic climate has made it impractical, or at least extremely risky, for cable television service providers to replace their existing base of possibly tens or even hundreds of thousands of receive-only cable set-top converter boxes with computerized transmit-and-receive cable modems. However, such two-way cable modems are needed to provide subscribers with interactive television and a real-time, high-speed link to information services, while simultaneously providing conventional television programming thereon.

At the same time, it is certain that many cable subscribers simply want low-cost television programming. This leaves cable providers with the dilemma of losing subscribers if they raise prices in order to upgrade their systems, or being left behind by new competitors and competing technologies if they do not. One solution is to convert to new technologies gradually. For example, conventional television programming can now be transmitted over cable in a digital format, enabling transmission of many more (virtual) channels on the same bandwidth cable. Rather than force all subscribers to convert to the digital format, however, cable service providers will transmit a mixture of analog and digital signals for a period of time. This reduces the total number of channels transmitted to less than the maximum possible allowed by the bandwidth, but allows the cable companies to keep subscribers who are reluctant to convert.

Regarding information services on the television, a number of low-cost devices presently exist for allowing access to information services using a television set instead of a computer monitor. However, these devices do not approach interactive television, but instead do little more than use the television display as an inexpensive monitor while connecting to the information service via a conventional, telephone-based modem. As a result, with such a device there is no direct connection between programs transmitted to a viewer and interaction with web pages or the like about those programs. For example, a viewer of a home shopping channel wishing to purchase a displayed item cannot simply do so with the low cost-device, but instead must first connect to the information service (e.g., the Internet) via the user's service provider, find the shopping channel's web-site (if one exists), find the appropriate item (if available on the web-site) and then place the order. Most significantly, these devices tie up a telephone line for long periods of time, a significant drawback in many households.

OBJECTS AND SUMMARY OF THE INVENTION

Accordingly, it is a general object of the present invention to provide an interactive entertainment and information system using a cable television set-top box.

A related object is to provide the above system using existing digital set top boxes without need for modification of the hardware therein.

Another object is to provide a system as characterized above that enables a user to hyperlink between pages of information.

Yet another object is to provide a system of the above kind that facilitates an interactive relationship between transmitted programming and information pages related thereto.

A related object is to allow a user to submit form information, including forms related to a viewed television program.

In accomplishing those objects, another object is to provide a simple to operate, low cost information service to those users desiring same.

Still another object is to provide a system of the above kind that is flexible and extensible.

Briefly, the present invention provides an interactive television system including a head-end having means for injecting video information into a transmission medium. The video information includes a plurality of page images. Detection means at a subscriber end, such as hardware and software in a cable set-top box in conjunction with a remote control device, detect a first request to display one of the plurality of page images. Upon detection of the first request, the box selects and displays the page image, wherein the page image includes at least one active location, such as a link, having an action corresponding thereto. The detection means further detects a second request to take the action that corresponds to the active location, whereupon the action is taken. For example, a new page might be displayed when a link is selected. The video information includes meta-data or the like associated therewith for providing information about the active location and the action to be taken upon selection thereof. Other actions may include those related to form entry and submission of form data.

Other objects and advantages will become apparent from the following detailed description when taken in conjunction with the drawings, in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram representing a cable television system in which the present invention may be incorporated;

FIG. **2** is a block diagram representing a head-end of a cable television system in accordance with the present invention;

FIG. **3** is a block diagram representing a digital set-top box for receiving cable transmissions at the subscriber end of the cable television system of FIG. **1**;

FIG. **4** is a diagram of an exemplary remote control device for providing user-commands to the digital set-top box of FIG. **3**;

FIGS. **5A–5B** comprise a flow diagram representing the general steps taken by a browser to access an information service in accordance with one aspect of the present invention;

FIG. **6** is a representation of an exemplary page image having page elements thereon;

FIG. **7** is a flow diagram representing the general steps taken by a browser to display a page;

FIG. **8** is a representation of an exemplary page image having form elements thereon;

FIG. **9** is a representation of the logical organization of program elements, including page images, within programs;

FIG. **10** is a block diagram representing server components for injecting page information into the cable transmission;

FIG. **11** is a flow diagram representing the general steps taken to construct a carousel of pages; and

FIG. **12** is a block diagram representing the authoring of a page for the information service.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

I. Overview

A basic feature of the Information Service architecture described below is the ability to deliver a stream of MPEG still images, or image pages, to a cable user (subscriber). Associated with these high-quality pages is a set of meta-data describing links to other pages along with a limited set of side effects and actions. The user can select and activate the links with a hand-held remote control, thereby interacting with the pages of information displayed on the television screen.

II. The Head-End

Turning to the drawings and referring first to FIG. **1**, there is shown a system generally designated **20** into which the present invention may be incorporated. In general, the system **20** transmits signals comprising video and audio information from a cable head-end **22** onto a coaxial cable transmission medium **24**, the signals being received at a plurality of subscriber locations $\mathbf{26}_1$–$\mathbf{26}_n$. Signal boosters (not shown) may be provided for amplifying and distributing the signals to the plurality of locations. At least one of the subscriber locations, such as the location $\mathbf{26}_1$, includes a digital set-top box **28** or the like equipped to convert the transmitted signals into signals capable of being received by a standard (e.g., NTSC) television set **30** for displaying video images and/or outputting audio to a consumer end-user. Of course, the present invention does not require coaxial cable as the physical transmission medium, as signals can alternatively be transmitted over any transmission medium, including wireless means such as so-called "wireless cable" broadcasts, digital satellite communication, and so on.

As shown in more detail in FIG. **2**, the head-end **22** includes a head-end LAN **32**, including a download server **34** and an access control server **36**, for controlling the operation of the head-end **22** via an Ethernet connection **38** to the various components therein. One of the components connected to the LAN is a satellite integrated receiver/transcoder **40**, which, as is known, receives and encodes digital television programming signals such as originating from microwave broadcasts received via a satellite antenna (dish) **42**. One such receiver/transcoder **34** is manufactured by General Instrument Corporation, Model No. IRT 1000, and outputs twenty-seven megabits per second (27 Mb/s) MPEG2 transport streams modulated onto a forty-four megahertz intermediate frequency carrier. In turn, the MPEG2-encoded transport streams are received by an intermediate frequency to radio frequency (IF/RF) up converter **44**, (General Instrument Corporation, Model No. C6U), which modulates the streams onto six megahertz (analog) channels and injects the signals onto the cable transmission medium **24**. With multiplexing of multiple, packetized digital signals per six megahertz analog channel, hundreds of digital channels may be injected onto a single coaxial medium, although at present a mixture of analog and digital channels will likely be transmitted.

According to one aspect of the invention and as described in more detail below, in addition to conventional television programming, the cable head-end **22** outputs a plurality of still images, or information pages to information service subscribers. To this end, the cable head-end **22** includes an information server **46**. The primary function of this head-end server **46** of the Information Service is to receive and store page data and carousel management information from content providers, and then to inject the page images onto the local cable system, i.e., produce a real-time carousel data stream for modulation onto a six megahertz channel of the transmission medium **24**. The server **46** may have the page image data permanently stored therein, but typically will regularly download at least some of the page image data from an external data source **48** such as the Internet. As described below, the subscriber can interact with the digital set-top box **28** to selectively display the various page images on the screen of the television set **30**.

More particularly, the information service server includes a carousel delivery application **49** for delivering a carousel **50** (FIG. **2**) of rendered HTML page images to the set-top box **28** along with meta-data for each page. Each page image consists of a single frame MPEG2 video sequence that is capable of being decoded by an MPEG video decoder **52** in the set-top box **28** (FIG. **3**). The meta-data for each page describe the structure and contents of the page image. As described above, the carousel **50** of page images and meta-data are delivered to a client set-top-box **28** as a standard MPEG2 Transport Stream, broadcast in-band over a six MHz NTSC channel.

The preferred head-end interactive information server **46** comprises a rack-mounted personal computer, including an Intel Corporation P6 200 megahertz (or better) central processing unit, 128 megabytes of RAM, a two gigabyte or larger hard disk drive, and an ISDN or better connection to the external data source (Internet) **48**. The server **46** also includes an Ethernet connection **38** to the local head-end LAN **32**, and a TAXI (serial protocol) interface card **54** (FIG. **2**) along with an appropriate TAXI driver **55** therefor. The operating system is Windows NT Server 4.0. Although only one such server **46** is necessary for a typical head-end **22**, two information service servers may be installed in a head-end for purposes of increased reliability based on redundancy. For redundant systems, the NT Servers will run Microsoft Clustering software.

As shown in more detail in FIG. **2**, to inject the signals, the information server **46** is connected via its taxi interface card **54** (on copper) to an in-band data modulator **56** connected to an intermediate frequency to radio frequency upconverter **58**.

The preferred in-band data modulator **56** is manufactured by General Instrument Corporation, model number IM-1000, which accepts 27 Mb/s transport streams from the information service server **46** and encodes and modulates those signals to a 44 MHz intermediate frequency. As with the redundant servers, there may be two IM-1000 in-band data modulators.

A second IF/RF Upconverter **58** (General Instrument Corporation, Model No. C6U) converts the 44 MHz intermediate signal to an RF signal and injects the RF signal into the cable system's transmission medium **24**. In this manner, the 27 Mb/s MPEG2 transport streams containing page images originating from the information server **46** are modulated onto a six megahertz analog channel for reception by subscribers.

Lastly, the head-end **22** includes an out-of-band data multiplexor **60** (General Instrument Corporation, Model No. OM-1000) which is a required component of virtually any head-end **22**. As is known, the out-of-band data multiplexor

60 may be used by the present information service to communicate information service-related data to the cable network. This data, including channel maps and possibly email as described below, is routed through the cable nation data center, and is thus only indirectly related to the operation of the information service head-end 22.

III. The Subscriber End

A. HARDWARE

Turning to the subscriber end of the system 20, as best shown in FIG. 3, the preferred set top box 28 is a digital set top box manufactured by General Instrument Corporation, Model No. DCT-1000. Although not necessary to the invention, it is significant that with this particular set-top box the hardware is unmodified, so that the existing base of publicly distributed set-top boxes may implement the Information Service without requiring upgrade servicing or replacement. However, to provide the service in accordance with the present invention, the operation of the box 28 is modified by additional software downloaded thereto. Such software includes a browser 62 which communicates with an operating system 64 of the box 28 by placing calls through an application programming interface (API) 66, as described in more detail below.

As shown in FIG. 3, the digital cable box 28 includes an in-band tuner 70 and an out-of-band tuner 71, along with appropriate demodulators 72 and 73, respectively. A microprocessor 74 controls the tuning operation of the tuners 70 and 71 based on commands received from a subscriber via an input device such as a keypad or an infrared remote control device 76, as described below. To this end, the set-top box 28 includes an infrared sensor 78 connected to an infrared receiver 80 which provides the command signaling information to the microprocessor 74. A memory system 82 includes the VRTX operating system 64 stored therein, and preferably comprises a combination of volatile dynamic RAM 84 and non-volatile RAM (NVRAM) 86.

In accordance with digital broadcasts wherein digitized channels are multiplexed as data packets onto a six megahertz analog channel, the set-top box 28 also includes at least three packet identification (PID) filters 88–90 to extract the appropriate encoded data packets for a user-selected digital channel. Based on the user-selected display, audio and other requirements, the microprocessor 74 writes an identification value to each of the PID filters 88–90, whereby the filters 88–90 pass only those packets corresponding to that value. As shown in FIG. 3, one of the PID filters, filter 88, provides the filtered packets to an audio decoder 92 which decodes the digital audio data (encoded according to the AC3 format), while another PID filter 90 provides filtered packets (MPEG2 encoded) to the video decoder 52.

As can be readily appreciated, in addition to line-level audio and video outputs, the resulting video signal may be output from the set-top box 28 with separate luminance and chrominance signals (SVHS format). As is typical, the set-top box 28 may also contain a modulator (not shown) for combining the audio and video signals onto a modulated carrier channel such as channel 3 or 4, for compatibility with television sets not having separate audio and video inputs.

A third PID filter 89 is provided to extract in-band and out-of-band data directed to the operation of the set-top box 28. A packet processor 94 handles those packets. The set-top box is also equipped with an on-screen display frame buffer (OSD) 96 capable of superimposing alphanumeric characters, other symbols and bitmap graphics over a displayed image. To accomplish this superimposition, an overlay 98 is provided to appropriately combine the video outputs of the video decoder 52 and the OSD 96.

The cable box 28 functions when the user provides an appropriate and valid command to the cable box 28. For example, in response to a digital channel selection command, the microprocessor tunes the in-band tuner 70 to an appropriate analog channel based on the digital channel selected by the subscriber. If a digital channel was selected, a table or the like stored in the memory 82 determines the analog channel that carries the digital channel's packets, along with the packet identification numbers corresponding to that channel, for writing into the PID filters 88 and 90. Once the PIDs have been written, the audio and video decoders 52 and 92 will receive the appropriate packets and decode and output appropriate signals. As described below with reference to the present invention, some of the packets will include page images associated with the information service.

The subscriber also will be provided with an input device, such as the hand-held remote control 76 best shown in FIG. 4. The preferred input device 76 includes four directional (up, down, left and right cursor) buttons, 100_1–100_4 respectively, and a "SELECT" button 102. The remote control 76 may include a dedicated button, chosen as the "A" button 104 of FIG. 4, which may be used to enter the information service in one alternative scenario described below. The preferred remote control input device 76 will also provide the normal complement of TV-related buttons including a numeric keypad 106, volume adjustment, channel adjustment, mute and so on. Other buttons such as those for control of a videocassette recorder also may be provided. The remote control is preferably wireless, e.g., an infrared or RF-based remote control, but of course alternatively may be wired. Moreover, alternate input devices need not be remote, but may for example, be provided as a keypad (not shown) on a set-top box.

B. THE BROWSER

In accordance with one aspect of the invention, the user utilizes the set-top box 28 to enter and interact with the Information Service. To this end, the browser 62 has been downloaded into the memory 82 of the set-top box 28, along with APIs 66 for interfacing the browser 62 to the operating system 64 of the set-top box 28. The operation of the browser 62 is generally described herein with reference to the flow diagram of FIGS. 5A–5B.

In one scenario, when the subscriber, via the remote control 76, tunes to a specified (and otherwise unused for programming) channel reserved for the Information Service, the browser 62 provides the subscriber with a page image having page elements displayed thereon including links to other information. The page images are obtained from the external data source 48, ordinarily the Internet, and are Hypertext Markup Language (HTML) pages provided by a third party that have been converted to an intermediate format as described below. For example, television stations, advertisers, pollsters and the like may obtain rights to have one or more page images transmitted on the cable medium 24 for potential viewing by the consumer end-user.

As shown by steps 500–502 of FIG. 5A, the initial page image is preferably a default (home) page that is displayed when this particular Information Service channel is selected. Another page based on information known to the system, such as a page corresponding to the previous channel being viewed by the user, may be displayed when the user enters this channel. Although not necessary to the invention, it is feasible that more than one such channel may be provided for entering the information service, with a potentially different initial page for each channel.

By way of example, the image 108 of FIG. 6, including a number of links 110_1–110_5, may be the default page

displayed when channel **99** is selected. For visibility purposes, the cable box **28** utilizes the on-screen display (OSD) **96** to draw focus (e.g., denoted in FIG. **6** by the visible ring, i.e., box **112** or the like) on one of the links when the page **108** is displayed. Other elements which can be focused are shown in FIG. **6** as being surrounded by dashed boxes (not ordinarily visible to an actual user). As shown by step **504**, the browser **62** then waits for a command from the user. One such command is entered as the user presses one of the directional cursor buttons $\mathbf{100}_1$–$\mathbf{100}_4$ on the remote control **76** (FIG. **4**).

As detected by step **504** and **506**, when the user presses a directional cursor button $\mathbf{100}_1$–$\mathbf{100}_4$, at step **508** the browser **62** reads the meta-data associated with the displayed page to determine how to adjust focus, (described below), and then draws the focus at the next location at step **510**. In this manner, the user tabs through the links $\mathbf{110}_1$–$\mathbf{110}_5$, changing the focus until a desired link is focused. Then, using the SELECT button **102** (FIG. **4**) of the remote control **76**, (as detected by steps **504** and **512** of FIG. **5**A), the user commands the browser **62** to take an action associated with the focused link. Examples of some possible actions are represented in FIG. **5**B, and include hyperlinking to another display page, filling in a check-box, submitting a form, and so on as described in more detail below.

If the user pushes the "A" button **104** on the remote control **76**, step **514** detects this command. Since the user entered the Information Service via the service's specific channel, step **516** returns the user to step **502** which displays the default page of the Information Service. In other words, pressing the "A" button in this scenario returns the user to the home page of the service. The user exits the service in this scenario by entering another channel, as detected by step **518**.

In another scenario, the user is watching programming on a specific channel that participates in the Information Service. If the user pushes the "A" button **104** on the remote control **76** (FIG. **4**) at any time while viewing such a participating channel, as detected by step **500**, an initial page image (the "top-level" page) corresponding to that channel is displayed (step **520**), along with links to other pages. For example, (assuming station participation), a user viewing ESPN will be shown an ESPN top-level page when the "A" button **104** is pressed. As before, if the top-level page provides links, the user can select from among those links and jump to other pages from that top-level page, but in this scenario those links ordinarily connect to only a small number of pages that are recognized as being "tied" to the original video channel. The user can exit the service by changing to another channel (step **518**), or by again pushing the "A" button **104** (step **514**). Note that in this scenario, step **516** does not display the home page when the "A" button is detected, but instead branches to step **522** where the set-top box is returned to the previous channel and the service is exited.

Moreover, if the service was entered via the "A" button **104** (FIG. **4**), the act of changing the channel resets and exits the Information Service. For example, if a page image page corresponding to a participating channel is being displayed, and the user changes to another channel but then returns to the original channel, the video program of the original channel will be displayed, not the page corresponding thereto.

Note that if another command is entered, (e.g., a PLAY command corresponding to a video recorder), the command can be handled by the browser **62** as desired. For example, the command can be treated like a channel selection (exiting the service) or just ignored. Steps **524**–**526** are shown to generally represent receipt of such a command.

In any event, if the user selects a page in the service and does not immediately select a link, it is possible for the page image to be updated at a frequency equal to the maximum initial latency of the carousel **50**. At present, a nominal maximum latency for an image to cycle back for display is about eight seconds, but this is variable depending on the amount of carousel bandwidth dedicated to a given page. More particularly, a frequently accessed page may be placed in the carousel **50** more than once at spaced-apart locations to reduce the latency for that page by increasing its frequency therein. For example, the service home page may be placed in the carousel **50** four times, providing a maximum latency of approximately two seconds for that page.

To facilitate the user interaction, once the service has been entered and a page is displayed, (step **502** or step **520**), using the OSD **96**, the browser **62** in the digital set-top box **28** draws focus on the first page element listed in a FocusDescriptor list in the meta-data (described below) associated with that page. The geometry (shape) of the indicator (i.e., the focus ring) drawn by the OSD **96** is also determined by information in the FocusDescriptor list. It is also feasible to specify the color, shading and the like of the focus ring in the meta-data.

As described above, the user may change the focus via the direction buttons $\mathbf{100}_1$–$\mathbf{100}_4$ on the remote control **76**, whereby the focus is re-drawn on the appropriate element (step **510**). More particularly, the order of changing focus in response to a directional button, i.e., the focus chain, is specified by the focus descriptors that are carried in the page meta-data. The focus descriptors explicitly specify where focus should be next drawn when the user depresses the up, down, left or right buttons $\mathbf{100}_1$–$\mathbf{100}_4$ on the remote control device **76**. For remote controls lacking direction keys, a single button can be used to tab through the links, in which event the order in which the focus descriptors occur in the focus descriptor list determines the focus chain.

In either arrangment, while focus is rendered on a particular element, the user may select that element by pressing the SELECT button **102** on the remote control **76** (step **512**), or an equivalent button (e.g., ENTER button) available on other devices. Regardless of how selected, when the element is selected, the browser **62** takes an appropriate action. The following table summarizes the actions taken by the browser for various types of page elements.

Element	Tag	Drawing Required	Browser action
Text Anchor	<A HREF=url> anchor text </A>	Focus	On selection jump to page indicated by URL.
Image Anchor	<A HREF=url> <IMG SRC=url> </A>	Focus	On selection jump to page indicated by URL.

-continued

Element	Tag	Drawing Required	Browser action
Client Image Map	<AREA SHAPE=shape COORD=coord HREF=url>	Focus	On selection jump to page indicated by URL.
Automatic Hyperlink	<META HTTP-EQUIV= REFRESH CONTENT="delay; URL=url">	None	Jump to page indicated by URL automatically after delay (in seconds).
Guide Hyperlink	<A HREF=url> GUIDE </A>	Focus	On selection pass the URL string to the GUIDE.
One Line Text	<INPUT TYPE=TEXT>	Focus "0"–"9"	Echo numerals when they are entered on the keypad. No action on selection.
Password	<INPUT TYPE=PASSWORD>	Focus "*"	Echo "*" when numerals are entered on the keypad. No action on selection.
Radio Buttons	<INPUT TYPE=RADIO>	Focus Check	On selection mark the button as checked and clear other buttons that are part of the button group.
Check-Box	<INPUT TYPE=CHECKBOX>	Focus Check	On selection mark the checkbox as checked.
Submit	<INPUT TYPE=SUBMIT>	Focus Border	On selection post the form query string for upload. Jump to the URL specified by the DESTINATION attribute in the <FORM> tag.
Reset	<INPUT TYPE=RESET>	Focus Border	On selection reset the form elements.
Image	<INPUT TYPE=IMAGE>	Focus	On selection Post the form query string for upload. Jump to the URL specified by the DESTINATION attribute in the <FORM> tag.
Hidden	<INPUT TYPE=HIDDEN>	None	Not selectable

To select and display a carousel page, the browser **62** generally follows the steps set forth in the flow diagram of FIG. **7**. First, as represented by step **700**, the browser determines a page to display, i.e., a default page or a linked page. At step **702**, the browser **62** references meta-data associated with that page to determine the page group to which that page belongs, and the digital channel corresponding thereto. As described in more detail below, a one-to-one mapping exists between a page group and a digital channel, and nine-to-one mapping exists between carousel pages and digital channels.

Cable boxes typically provide APIs **66** to control the tuner and selection of program elements in the MPEG2 stream. The browser **62** uses such an API (in the APIs **66**) to specify the digital channel corresponding to the appropriate page group on the carousel **50**, whereby the microprocessor **74** in the set-top box **28** tunes the in-band tuner **70** to the appropriate carrier (step **704**) and selects the program (Service) corresponding to the desired page group (step **706**). Note that the mapping between digital channels and carousel page programs is fixed. Next, at step **708**, the browser **62** selects the program element (service component) containing the carousel page for display, and uses an API (or the like) to select the Video Program Element for display (step **710**). As a result of this step, the PID is written to the PID filter **90** and the page image is decoded and displayed by the hardware.

At step **712**, the Browser **62** selects the program element that carries the meta-data for the page group. Cable boxes may provide an API for this function, this time to acquire an identifier for the page meta-data. At step **714**, the browser **62** reads the page meta-data, which, as described in more detail below, includes a record for each element on the page. Each element record includes information such as the geometry of focus for that element and other information specific to that element. For anchor elements, (described below), the record includes the digital channel and program element for the page to which the link refers. Note that the carousel **50** contents are completely described by the carousel meta-data, which is carried in-band. No out-of-band or backchannel data transmission is required to navigate between pages in the carousel **50**.

After the meta-data has been read (step **714**) by the browser **62**, at step **716**, the browser **62** draws focus on the default link as specified by the page meta-data. At this time, the page is ready for user-interaction. For example, as described above, the browser **62** draws focus on other links as they are tabbed to by the user, according to the focus chain specified in the page meta-data.

The browser **62** supports a subset of HTML anchors. The processing of each type of anchor is described below. The following table gives a brief summary of anchors supported by the Information Service.

Element	Tag	Support	Attribute Support	Limits	Drawing Required	Notes
Text Anchor	<A HREF=url> anchor text </A>	Yes			Focus	Specifies text hyperlink. HREF indicates destination.

-continued

Element	Tag	Support	Attribute Support	Limits	Drawing Required	Notes
Image Anchor	<A HREF=url> <IMG SRC=url> </A>	Yes			Focus	Specifies image hyperlink. HREF indicates destination.
Client Image Map	<IMG SRC=url USEMAP=map> <MAP NAME=map> <AREA SHAPE=shape COORD=coord HREF=url> </MAP>	Yes			Focus	Specifies client-side imagemap. HREF indicates destination.
Server Image Map	<A HREF=CGIurl> <IMG SRC=url ISMAP> </A>	No				Use client-side image map.
Automatic Hyperlink	<META HTTP-EQUIV= REFRESH CONTENT="delay; URL=url"> SLIDESHOW=YES	Yes				Hyperlink specified by URL is automatically taken after delay (in seconds).

The focus geometry for anchors and areas in imagemaps are specified in the page meta-data. The browser **62** uses this information to draw focus on hyperlinks and to select other pages for display when hyperlinks are selected. The page meta-data also specify the focus chain. At present, the browser **62** handles text anchors, image anchors, and areas within imagemaps identically. Thus, when the user moves focus to a hyperlink, the browser **62** simply draws focus, using the OSD **96**, on some region of the screen as indicated by geometry information in the page meta-data. When the user selects the link, the browser **62** changes the display to the destination page as indicated by the link data in the former page's meta-data.

At present, it is not possible to specify explicitly that an anchor within a page should receive initial focus. Instead, the first anchor or imagemap in the HTML source will be the element that receives initial focus when a page is first displayed. Within an imagemap the order of the <AREA> tags will determine the order of focus within that imagemap. The order of the <IMG> tag within the HTML source will determine where the imagemap anchors will occur in the focus chain. Note that it is possible to edit the focus chain for a page at page conversion time, (described below), in order to achieve results that cannot be generated simply in an HTML source.

Automatic hyperlinks are allowed, in which after the page is displayed and the delay (specified in meta-data) has elapsed, the browser **62** displays the page to which the hyperlink refers. As before, the page meta-data of the former page includes information that the browser **62** uses to select the destination page for display. Nothing is rendered in response to an automatic hyperlink.

A server-side slideshow may also be indicated, in which a series of still images is sequentially displayed on the same page. Unlike a client-side autolink to another page, (which can produce a similarly-appearing client-side slideshow using multiple pages), the page change for a server-side slideshow takes place at the server side. Via the slideshow tag, the server **46** inserts a new page image in place of the old with each new cycle of the carousel **50**. On the client side, the displayed slideshow page autolinks to itself to reacquire page meta-data. The slideshow feature enables the injection of a series of still images with appropriate meta-data, while only using a single page of the carousel **50**. Moreover, the server **46** may obtain the series of images in advance, eliminating real-time downloading thereof. By way of example, a real estate entity may use this feature to present a series of images of new homes with accompanying meta-data, but only use one page of those available in the carousel **50**. Note that a slideshow can present an image for any multiple of the carousel revolution time, e.g., eight seconds, by replicating images in a series as desired.

Moreover, this feature can be used in conjunction with client-side autolinks (using more than one page) to produce other update intervals. Indeed, by rebuilding the carousel, animation may be accomplished. To this end, a plurality of still images with slight changes from image to image will appear as animated movement. If a single-page slideshow, the animation may appear somewhat slow depending on the latency (e.g., a movement every eight seconds). However, with autolinks to multiple pages, the animation may be sped up. Note that MPEG2 is arranged to transmit change information between frames, and thus not much bandwidth is required to accomplish animation. Other image enhancements such as brightening a focus ring, adding images together and so on may be performed by rebuilding the carousel, taking advantage of MPEG2 features wherever possible.

In keeping with the invention, the browser **62** supports a subset of HTML forms with a few extensions. The processing of each of the form elements is described below. The following table gives a brief summary of forms supported by the Information Service. Only those attributes listed are supported at this time.

Element	Tag	Support	Attribute Support	Limits	Drawing Required	Notes
Action	<FORM ACTION=url>	Yes	ACTION METHOD ENCTYPE DESTINATION=url			Form submission is standard syntax but is not sent via HTTP until the STB is polled.
One Line Text	<INPUT TYPE=TEXT>	Yes	NAME SIZE ALIGN DISABLED	Numeric Only MAXLENGTH == SIZE	Focus "0"–"9"	Intended for credit card number and quantity entry only. No editing of field. User must use Reset to clear field.
Multiple Line Text	<TEXTAREA>	No				
Password	<INPUT TYPE=PASSWORD>	Yes	NAME SIZE ALIGN	Numeric Only MAXLENGTH == SIZE	Focus "*"	Intended for PIN entry only. No editing of field. User must use Reset to clear field. Characters are not echoed.
Drop Menus	<SELECT>	No				Use radio buttons
Text Menus	<SELECT MULTIPLE>	No				Use radio buttons
Graphic Menus	<SELECT MULTIPLE SRC=url>	No				Use radio buttons
Radio Buttons	<INPUT TYPE=RADIO>	Yes	NAME VALUE CHECKED ALIGN		Focus Check	
Check-Box	<INPUT TYPE=CHECKBOX>	Yes	NAME VALUE CHECKED ALIGN		Focus Check	
Range	<INPUT TYPE=RANGE>	No				Use type=TEXT
Submit	<INPUT TYPE=SUBMIT>	Yes	NAME VALUE ALIGN		Focus Border	
Reset	<INPUT TYPE=RESET>	Yes	NAME ALIGN		Focus Border	Allows simple edit control for text entry. User can only reset all fields.
Image	<INPUT TYPE=IMAGE>	Yes	NAME SRC ALIGN		Focus	
Hidden	<INPUT TYPE-HIDDEN>	Yes	NAME VALUE AUTOSUBMIT		None	Information Service specific VALUEs will return textual user information (Name, Address, Phone)
Button	<INPUT TYPE=BUTTON>	No				Use type=SUBMIT
Scribble	<INPUT TYPE=SCRIBBLE>	No				

Using forms, a page can have simple data entry links, including check boxes, radio buttons, and numeric entry (text boxes). By way of example, FIG. **8** shows an exemplary page **116** including text boxes 118_1–118_4, radio buttons 120_1–120_4, and a check box **122**. In addition, the page **116** includes a button **124** for submitting an order, a "RESET" button **126** for clearing entries, and a hyperlink element **128**. In FIG. **7**, focus is shown (by the solid rectangular ring) as being on the "PURCHASE NOW" submit button **124**, while elements which can be focused are shown surrounded by dashed boxes (not ordinarily visible to an actual user).

One such form is a guide form, which is processed in real time by the programming guide to take an appropriate action upon user submission thereof. As described below, depending on the current time, the programming guide allows a user to link to a currently available program, add the program to a timer for timed viewing thereof, and/or prompt the user to purchase a pay-per-view event.

Another type of form is for purchasing merchandise or the like. To perform a purchase with this type of "store-and-forward" form, the user tabs through the various elements to select and enter appropriate information. The entered attributes (e.g. color, size, style, quantity, payment method and so on) are temporarily stored in the memory **82**. Once the appropriate data is entered in this manner, the user selects the "PURCHASE NOW" link **124**, which stores the selected information as data **130** (FIG. **3**) (preferably in the non-volatile memory **86**) in the set-top box **28**. At a later time, the purchase information will be collected from the set-top box **28** with a polling procedure, whereby each box in the cable system **20** is directed in turn to call a polling server **132** via its modem **134** (FIG. **3**), typically late at night.

More particularly, in a preferred embodiment of the present invention, there is no real-time interaction with a web server. As a consequence, form submission must be emulated. The meta-data for a page image includes all the necessary information required to generate a standard URL query string. When a form is submitted, data including the ACTION URL, the query string, and METHOD are stored as data **130** (preferably in NVRAM **86**) in the set-top-box **28** as an ASCII string. These data are formatted exactly like a standard URL query string. The METHOD, GET or POST

is appended to the form query string. In other words, a form element named _METHOD is appended to every form query string automatically. For instance, "&_METHOD=GET". An upload forms server **132** of the Information Service removes this sub-string when the form data are processed.

When the set-top-box **28** is polled via its modem **134** for the return data (typically once per day), the form data for all such forms which have been submitted are returned to the upload forms server **132** of the Information Service. The server **132** processes the form data and generates the appropriate HTTP GET or POST requests to the specified ACTION URL. Of course, user information, including name, billing and mailing addresses, and credit card type and number, will need to be collected from each user before a purchase can be made. This information can either be held remotely by the polling server **132**, or stored within the set-top box **28** to be sent with each purchase. If this information is not available or valid, then the "PURCHASE NOW" button will link to an "error" page in the carousel, which will direct the user to call the service provider.

No immediate confirmation of the purchase will be available. The CGI scripts for Information Service forms cannot return HTML documents to subscriber-users since the contents of the carousel are broadcast and not unicast. However, the CGI forms script can send Email to the user in response to a forms query. To this end, the Email address of a particular set-top-box can be returned with the form data by using the Magic Element Name _EMAIL in a HIDDEN form elements (described below). Thus, an e-mail confirmation notice may be sent to the user, as described below with reference to the e-mail scenario.

The focus geometry for form elements is specified in the page meta-data, described below. As before, the browser **62** uses this information to draw focus on the form elements. At present, it is not possible to specify that a particular form element has the default focus for that form. Instead, the first form element in the HTML source will be the first element that receives focus when the user tabs to the form. It is also not presently possible to specify that a particular form element has the default focus for the page in which that element is contained. If, however, the form is the first element in the HTML source, then the first element in that form will receive the initial focus.

Note that for high-quality imaging, (relative to the low quality provided by the OSD **96**), the face of a submit button (such as the "PURCHASE NOW" link **124**) is rendered at page conversion time as part of the compressed page image. The button borders, however, are rendered at display time using the OSD **96** of the set-top-box **28**. The page meta-data includes information that the browser **62** uses to draw button focus as well as drawing button borders.

As a result, when the user selects a submit button, the browser **62** directs the OSD **96** to draw the button borders such that the button appears to be depressed and released. As generally shown in FIGS. **5A–5B**, when the user selects a submit button (step **512**, FIG. **5A** and step **530**, FIG. **5B**), the browser **62** then queues the appropriate form query string (step **532**) for upload as described above, (or for processing by the program guide as described below). If a DESTINATION attribute is specified on the containing form, (as detected by step **534**), then following the submit command the browser **62** will jump to the page corresponding to that destination (step **536**).

Similarly, a reset button face (e.g., "RESET" button **126**) is rendered at page conversion time as part of the compressed page image. The button borders are likewise rendered at display time using the OSD **96** of the set-top-box **28** so that the button appears to be depressed and released. The page meta- data includes information that the browser **62** uses to draw button focus as well as drawing button borders. When the user selects the reset button (step **538**), the browser **62** appropriately draws the button borders and restores the form elements to their default state (step **540**).

Any page that contains a text entry box or password box (described below) contains a reset button so that the user can clear errors. At present, the text entry and password boxes allow entry but do not support any other type of editing.

Any submit element, e.g., a "SUBMIT" button or a submit image element, is also rendered at page conversion time as part of the compressed page image. The page meta-data includes information that the Browser **62** uses to draw focus on the image. When the user selects the Submit Image, the Browser **62** queues the appropriate form query string for upload or processing by the guide. If a DESTINATION attribute is specified on the containing form then the Browser **62** will jump to that page.

Check boxes are also rendered at page conversion time as part of the compressed page image. Since a check mark may or may not appear based on the user entry, the check mark is rendered at display time using the OSD **96** of the set-top-box **28**. The page meta-data includes information that the browser **62** uses to draw box focus as well as drawing the check marks on the box. When the user selects a check box (step **542**), the browser **62** inverts the state of the box (step **544**) in the memory **82** and correspondingly changes the display by either drawing or erasing a checkmark (shown as an "X" in box **122** of FIG. **8**) on the selected checkbox. When the form is submitted, the boxes that are checked result in name/value pairs in the form query string.

Similarly, radio buttons are rendered at page conversion time as part of the compressed page image. The user-movable check (shown as a darkened circle in box $\mathbf{120}_2$ of FIG. **8**), however, is rendered at display time using the OSD **96** of the set-top-box **28**. The page meta-data includes information that the browser **62** uses to draw button focus as well as drawing check marks on the button. When the user selects a radio button (step **546**), the browser **62** sets the state of that button to selected while de-selecting others with the same name. At step **548**, the browser **62** also causes the OSD to draw a suitable checkmark on the button, and erase the checkmarks on any other buttons with the same name. When the form is submitted, the button that is checked results in a name/value pair in the form query string.

At present, a Text Entry Box only supports entry and echo of the numerals "0" through "9," and only a single font and pitch are supported. A Text Entry Box is rendered at page conversion time as part of the compressed page image. Echoed text is rendered at display time using the OSD **96** of the set-top-box **28**. The page meta-data includes information that the browser **62** uses to draw focus as well as information for drawing the characters.

When a text box is selected at step **512** (FIG. **5A**) and **550** (FIG. **5B**), the browser **62** draws focus on the text box. If the user depresses a number key on the keypad then at step **552** that number is displayed in the leftmost position of the text box and advances the text entry position to the next character position. No cursor is presently displayed to indicate the text entry position, and the next character is simply the rightmost unoccupied character position in the text box. When the user depresses another number key on the keypad, that number is displayed in the new character position and the text entry position is advanced. When the form is submitted a name/value pair is appended to the form query string.

Certain rules for text boxes presently apply. If the user attempts to enter more than characters then allowed in the SIZE field of the meta-data (described below), those characters are ignored. Moreover, for now a text box does not support any type of editing, and thus in order to clear errors, the user uses a reset button on the form. No default values (using the VALUE attribute) may be specified for text boxes. The Password Box is identical in function to the text box except that the characters are not echoed. Instead, an asterisk ("*") or the like is displayed at step **552** for every character that is entered.

The browser **62** also supports the DISABLED attribute, (seldom supported by legal HTML), which, when used in conjunction with the magic name "_TOTAL" can be used to generate a read-only text box which displays the total cost of a purchase transaction. For instance the tag <INPUT TYPE= TEXT NAME=_TOTAL SIZE=5> will generate a text box 5 characters in size in which the following result is displayed: _TOTAL=_QUANTITY*_PRICE+_SANDH, where _QUANTITY is presumably to be entered by the user in a text box (e.g., the text box $\mathbf{118}_4$) and _PRICE and _SANDH are specified using hidden text. Such a read only text box **136** is shown in FIG. **8**.

Hidden text results only in page meta-data. Nothing is rendered either at page conversion time or at display time. When a form is submitted Hidden Text results in a name/value pair being appended to the form query string.

Certain state information stored within the set-top-box may be returned as part of the form query string by using "magic element names" for the NAME of the hidden text. In such an event, no VALUE need be specified.

The Information Service of the present invention adds two attributes to the standard HTML syntax. More particularly, a first way in which the Information Service extends the standard HTML hidden text syntax is by adding the attribute AUTOSUBMIT to the tag <INPUT TYPE=HIDDEN>. When the user leaves a page with a form containing a hidden text element with the AUTOSUBMIT attribute, the form query string is queued for upload just as if the user had selected a submit button.

Second, the attribute DESTINATION has been added to the tag <FORM>. This tag causes the Browser **62** to jump to the specified page when the user submits the form. This allows the author to specify an HTML page that simulates the result of the HTTP GET on normal form submission (e.g., through CGI). This tag is unique to the Information Service system.

For any page, it is possible to include a link that tunes the set-top box **28** to a video channel rather than a link to another page. To this end, the Information Service facilitates interaction with a programming guide (e.g., PreVue). The exact behavior of the link depends on the type of video program that is selected. If the link selects a channel with no time (of day) associated therewith, or selects a channel with a time that indicates a currently available program, the set-top box immediately tunes to the selected channel. If the associated time is in the future, the user is presented with a programming guide pop-up window and prompted to add the program to a timer for timed viewing thereof. If the program is a pay-per-view event, the user is prompted to buy the event, and, if purchased, the browser **62** either tunes to the channel or adds the program to the timer.

To accomplish such actions, the Information Service provides access to functions in the programming guide through the use of forms. A form that accesses the guide does so by specifying an ACTION URL that is processed by the guide, that is, the guide looks like a web-server running a CGI script. One such guide is accessed using the URL "http://www.prevue.com/cgi/guide".

At page conversion time, this URL is recognized, and meta-data for the page indicate that the form query string should be submitted to the guide for processing. At page display time, when the form is submitted, the browser **62** submits the form query string to the guide rather than queue it for upload. To reduce meta-data size and simplify argument processing, the query string is preferably collapsed into a compact binary representation thereof at page conversion time. This syntax allows page image authors to access guide emulation (e.g., provided by PreVue) at author time. The following table sets forth the queries that are supported by the guide:

Function	Query Syntax	Description
TUNE	Function=TUNE &SourceID=program_name &Time=now	Switch to a television program
REMIND	Function=TUNE &SourceID=program_name &Time=some_future_time	Set a reminder for a television program
RECORD	Function=REC &SourceID=program_name &Time=some_future_time	Set a reminder for a television program
EMAIL HELO	Function=HELO	Opens EMAIL session
EMAIL QUIT	Function=QUIT	Closes EMAIL session
EMAIL DATE	Function=DATE &UpperLeft=xlocation+ylocation &LowerRight=xlocation+ylocation	Display current message RFC822 "Date" text at the specified screen location.
EMAIL FROM	Function=FROM &UpperLeft=xlocation+ylocation &LowerRight=xlocation+ylocation	Display current message RFC822 "From" text at the specified screen location.
EMAIL SUBJ	Function=SUBJ &UpperLeft=xlocation+ylocation &LowerRight=xlocation+ylocation	Display current message RFC822 "Subject" text at the specified screen location.
EMAIL BODY	Function=BODY &UpperLeft=xlocation+ylocation &LowerRight=xlocation+ylocation	Display current message RFC822 message body text at the specified screen location.

-continued

Function	Query Syntax	Description
EMAIL ACKS	Function=ACKS	Save the current message and increment the current message indicator.
EMAIL ACKD	Function=ACKD	Delete the current message and increment the current message indicator.
EMAIL ACKB	Function=ACKB	Save the current message and decrement the current message indicator.

IV. Page Images

As described above, the carousel of page images and meta-data are carried as a single MPEG2 Transport Stream, and each page image is a single MPEG2 video sequence consisting of a single I-Frame image of the rendered HTML page. As generally represented in the program map tables **138**$_1$–**138**$_n$ of FIG. **9**, each page image is carried as a single program element (i.e., Service Component) of a program (i.e., Service), and, (as limited by the DCT1000 set-top box), each program (Service) ordinarily consists of ten program elements. As further seen in the program map tables **138**$_1$–**138**$_n$ of FIG. **9**, nine of the program elements (i.e., a page group) contain page images, and the other program element contains the meta-data for the nine page images that make up the program. The meta-data are carried as the first program element within a program.

As also represented in FIG, **9**, each group of nine image pages in the carousel **50**, which correspond to a distinct program, is mapped to a single digital channel. Thus, (as described above with reference to FIG. **7**), in order to display a particular page, the browser **62** tunes to the appropriate digital channel and selects the appropriate program element (Service Component) for display. The browser **62** also selects the first (meta-data) program element so that it can render focus and execute links as directed by the user. The transport stream includes the appropriate PSI data to allow the page images to be selected and displayed.

According to digital transmission conventions, the program map table **138**$_1$–**138**$_n$ (i.e., Service Definition Table) maintained for each program contains **10** entries, and, including the PID which caries the program map table, thus uses **11** PIDs. A program association table **139** (Service Association Table) associates programs with program map PIDs and thus contains as many entries as the number of carousel pages divided by nine. The maximum number of page images that may be carried by a single carousel **50** is limited to 6700 pages. Note that if PCRs must be supplied to ensure that the set-top-box NTSC sub-carrier does not drift, then a single valid PCR stream will be multiplexed with the carousel data. The program map table for all programs references this PID as the PCR_PID.

Pages are delivered to the server in a two-stage process. First, a content provider transfers information (from an external source such as the Internet **48**) to the Information Service server **46**, the information including provider identification, a root URL of the content, and the date and time the content is to become valid. Any acceptable protocol for this transmission is feasible, including Hypertext Transfer Protocol (HTTP) or a private message protocol on top of TCP/IP.

Second, using HTTP, the Information Service server gathers the content at the specified URL and stores it for processing into the carousel **50**. The timing of this transfer may vary depending on several factors, including available storage at the server, number of pending transfers, date and time the content is to become valid and default actions specified by the cable operator. In any event, the content is transferred before the start time. For real-time updates (e.g. sports, scores or stock information), the same process is followed, but with an immediate start time. The system is designed to allow for low latency for both the information transferring and content gathering stages.

As generally represented in the block diagram of FIG. **10**, a conversion process **142** converts the page **140** into an intermediate page format **144**, where it is passed to a carousel builder **146**. One purpose of the intermediate page format is to reduce the amount of processing that needs to be performed by the processor **74** at the cable end. For example, rather than have the cable end processor dynamically calculate from an HTML page layout where to jump in response to a cursor key, the jump is pre-decided and stored in the intermediate page format.

The IPF file includes an image tag, a program information tag and a meta-data tag as described below. Note that the values correspond with fields within the meta-data as described herein.

Image Tag

<META NAME=PARAKEET_IMAGE CONTENT="Url">

where Url represents the URL of the MPEG video file, (e.g., www.content.com/Page1.M2V).

Program Information Tag

<META NAME=PARAKEET_PROGRAM_INFO CONTENT="
PROGRAM=Program ELEMENT=Element PACKAGE=Package
">

where PROGRAM=Program specifies the program_number or SourceID of the page. This is only relevant for quick pages, i.e., a page that is mapped as a channel, (for example, channel **100** might be the weather page). ELEMENT=Element specifies the program_element or ServiceComponentID of the page (only relevant for quick pages), and PACKAGE=Package specifies the package number for premium or tiered services.

Meta-data Tag

<META NAME=PARAKEET_METADATA CONTENT="
_PAGE_META_DATA TYPE=Type AUDIO=Url DELAY=Time
_ELEMENT_DESCRIPTOR TYPE=Type
_LINK_DESCRIPTOR Url
_FOCUS_DESCRIPTOR SHAPE=Shape COUNT=CoordCount UP=UE DOWN=DE RIGHT=RE LEFT=LE COORDS=X,Y,X,Y, . . .
_FORM_DESCRIPTOR NAME=Name VALUE=Value COORDS=X1,Y1,X2,Y2
FORM DESCRIPTOR FUNCTION=Function TIME=Time COORDS=X1,Y1,X2,Y2

">

The PAGE_META_DATA contains the per-page meta-data, wherein TYPE=Type indicates the type of the page (e.g. Hold, Reload, Autolink, and so on), AUDIO=Url identifies the URL of the background audio file (e.g., www.sounds.parakeet.com/classical.ac3), and DELAY= Time specifies the delay in seconds for Autolink pages. For each element on the page one or more of the following tags will be found, and will be grouped together such that the tags for a particular element occur together, (_ELEMENT_ DESCRIPTOR, _LINK_DESCRIPTOR, _FOCUS_ DESCRIPTOR, _FORM_DESCRIPTOR). Note that there are two formats for the _FORM_DESCRIPTOR. The following describes the information in the tag:

TYPE=Type—the element type (e.g. LINK, FORM, INPUT_SUBMIT, and so on),

Url—for elements that link the URL of the destination page (e.g. www.contentprovider.com/pageN.ipf),

SHAPE=Shape—shape of the focus, (e.g., CIRCLE, RECT, POLYGON),

COUNT=CoordCount—the number of coordinates for focus,

UP=UE—the element to receive focus on up-button press,

DOWN=DE—the element to receive focus on down-button press,

RIGHT=RE—th element to receive focus on right-button press,

LEFT=LE—the element to receive focus on left-button press,

COORDS=X,Y,X,Y, . . . —the Coordinates for focus,

NAME=Name—the NAME attribute of form element

VALUE=Value—the VALUE attribute of form element

COORDS=X1,Y1,X2,Y2—the coordinates of form element (e.g., the button corners),

FUNCTION=Function—the GUIDEFORM function (e.g. TUNE),

TIME=Time—the effective time for TUNE or REC function, and

COORDS=X1,Y1,X2,Y2—the coordinates for text field on GUIDEFORMs.

The following shows an example of an IPF file for a simple page only containing links:

```
<META NAME=PARAKEET_ORIGINAL_HTML
CONTENT="
<!--META HTTP-EQUIV="Refresh" CONTENT="5;
URL=page17.htm"--
>
<HTML><BODY SCROLL=NO BGCOLOR=BLACK
leftmargin=0 topmargin=0>
<IMG SRC=Weather1.jpg BORDER=0 USEMAP=
"#coords">
<MAP NAME="coords">
<AREA SHAPE=RECT COORDS="427,36,583,68"
HREF="page17.htm">
<AREA SHAPE=RECT COORDS="502,385,584,445"
HREF="page0.htm">
<AREA SHAPE=RECT COORDS="425,108,585,142"
HREF="page5.htm">
<AREA SHAPE=default HREF="page0.htm">
</MAP>
</BODY></HTML>
">
<META NAME=PARAKEET_IMAGE CONTENT=
"Page1.M2V">
<META NAME=PARAKEET_PROGRAM_INFO
CONTENT="PROGRAM=16129
ELEMENT=0 PACKAGE=0">
<META NAME=PARAKEET_METADATA CONTENT="
_PAGE_META_DATA TYPE=HOLD AUDIO=NONE
DELAY=0
_ELEMENT_DESCRIPTOR TYPE=LINK
_LINK_DESCRIPTOR page17.ipf
_FOCUS_DESCRIPTOR SHAPE=RECT COUNT=2
UP=2 DOWN=1 RIGHT=1 LEFT=2
COORDS=237,36,323,68
_ELEMENT_DESCRIPTOR TYPE=LINK
_LINK_DESCRIPTOR page0.ipf
_FOCUS_DESCRIPTOR SHAPE=RECT COUNT=2
UP=0 DOWN=2 RIGHT=2 LEFT=0
COORDS=279,385,323,445
_ELEMENT_DESCRIPTOR TYPE=LINK
_LINK_DESCRIPTOR page5.ipf
_FOCUS_DESCRIPTOR SHAPE=RECT COUNT=2
UP=1 DOWN=0 RIGHT=0 LEFT=1
COORDS=236,108,324,142
">
```

To build the carousel **50**, the carousel builder **146** generally follows the steps of FIG. **11**, whereby at step **1100** a program element (Service Component) is allocated from the program (Service) which will carry the page and its meta-data. Note that there is a one-to-one mapping between program elements and pages and a nine-to-one mapping between pages and programs. At step **1102**, the page image is then wrapped with the appropriate Transport Stream systems-layer syntax using the appropriate PID from the program map table (Service Definition Table). Then, at step **1104**, the page meta-data are wrapped in the appropriate private_message_stream syntax and Transport Stream syntax, using the appropriate meta-data PID from the Program Map Table.

At step **1106**, the meta-data (described below) for any pages that contain links to the (newly provided) page are modified to contain the program number and program element for this page. Similarly, at step **1108**, the meta-data for the newly-provided page is modified to contain the program number (SourceID) and program element (Service Component) for pages that are linked to by the page. The initial state of the link program number in the page meta-data is such that a Null page is indexed. The Null page resides on program number 1, and contains an image indicating that the page is not available. Lastly, as represented by step **1110**, when the carousel **50** is ready for transmission, the information server **46** periodically takes each carousel image stored in the **50** and injects the image onto the transmission medium **24**, a process which continues for the carousel **50** until page information therein is changed. As a result, pages are periodically available at the subscriber end as described above.

The carousel builder **146** maintains a carousel description that is modified by page update messages. Each time the carousel **50** changes, the carousel builder **146** creates a new carousel image (in server memory) from the carousel description and the pages (stored on disk). When the new carousel image is completed, it replaces the old carousel at the beginning of the next cycle. Note that in the case of pages that change each cycle, i.e., "slide show" pages, a new carousel must be built for each cycle.

V. Meta-Data

In order for the user to interact with the pages, the page images have meta-data and PSI data associated therewith in the Transport Stream. In general, the carousel page meta-data contain the information necessary for the browser **62** to render focus on the links on the page corresponding thereto. The meta-data also contain information for selecting a new

page image when the user selects a link, gather form input from a subset of Hypertext Markup Language (HTML) form elements and/or post form input for deferred processing.

The meta-data for a page group (of nine pages) are carried in a single program element consisting of private_stream_ messages. To associate the appropriate meta-data with its corresponding page, the meta-data is fragmented into multiple private_stream_messages such that the meta-data for a particular page is carried as a unique session. Each private_stream_message is capable of containing a predetermined number of bytes of meta-data. The meta-data for a page consists of four tables, identified herein as the page element table, the link table, the focus table, and the form table. These tables are described in more detail in the following sections.

The following table provides a high-level description of the information that is carried in the meta-data for various HTML constructs:

HTML Tag	Meta-data content	Description
<A HREF=url>	Focus Specifier Page Address	Describes how to draw focus on the link and the program_number and Program Element that carry the destination page
<AREA SHAPE=shape COORDS=coords HREF=url>	Focus Specifier Page Address	Describes how to draw focus on the link in a client-side image-map and the program_number and Program Element that carry the destination page
<FORM ACTION=url DESTINATION=url>	Transaction URL Page Address	Indicates where the form result should be posted. When the form is submitted, the DESTINATION page is displayed.
<INPUT TYPE=SUBMIT>	Focus Specifier Location Specifier	Describes how to draw focus and borders on the SUBMIT button and indicates that form should be posted when the button is selected.
<INPUT TYPE=IMAGE SRC=url>	Focus Specifier	Describes how to draw focus on the image and indicates that the form should be posted when the image is selected.
<INPUT TYPE=HIDDEN AUTOSUBMIT>		Indicates that the form should be submitted automatically when the user leaves the page.
<INPUT TYPE=HIDDEN NAME=name VALUE=value>	Form Element Name Form Element Value	Indicates that the name/value pair should be submitted with the form. Nothing is displayed.
<INPUT TYPE=HIDDEN NAME=magic_name>	Form Element Name	Same as above but the value is retrieved from the local user database based on the name.
<INPUT TYPE=CHECKBOX NAME=name VALUE=value>	Focus Specifier Location Specifier Form Element Name Form Element Value	Describes how to draw focus and draw check on checkbox. Indicates name/value pair to be returned if the box is checked.
<INPUT TYPE=RADIO NAME=name VALUE=value>	Focus Specifier Location Specifier Form Element Name Form Element Value	Describes how to draw focus and draw check on radio button. Indicates name/value pair to be returned if the box is checked.
<INPUT TYPE=TEXT NAME=name VALUE=value>	Focus Specifier Text Specifier Form Element Name Form Element Value	Describes how to draw focus and draw text on the text entry element and the initial value. Only numerals are supported. No edit control, user must use RESET to clear the field.
<INPUT TYPE=PASSWORD NAME=name>	Focus Specifier Text Specifier Form Element Name	Describes how to draw focus and draw text on the password entry element. Only numerals are supported. They are echoed as "*". No edit control, user must use RESET to clear the field.
<INPUT TYPE=TEXT NAME=_TOTAL>	Location Specifier Form Element Name	No focus is rendered on this element. It is output only and numeric only. The value is the product of the magic names _QUANTITY and _PRICE added to _SANDH.
<INPUT TYPE=RESET>	Focus Specifier	Describes how to draw focus on the RESET button and indicates that form elements should be reset when the button is selected.
<META HTTP-EQUIV= REFRESH CONTENT="delay; URL=url">	Page Address Delay	Causes the browser to jump to the specified URL after the specified delay automatically.

The page Meta-data contains all of the meta-data for a page, as described in the following table:

Syntax	No. of bits	Mnemonic
PAGE_META_DATA( ){		
ID	16	Uimsbf
Type	8	Uimsbf
Reserved	8	Uimsbf
Delay	16	Uimsbf
Reserved	16	Uimsbf
LinkOffset	16	Uimsbf
FocusOffset	16	Uimsbf
FormOffset	16	Uimsbf
for (i=0;i<LinkOffset;i++){		
LINK_DESCRIPTOR( )		
}		
for (I=0;i<FocusOffset;i++){		

-continued

Syntax	No. of bits	Mnemonic
FOCUS_DESCRIPTOR( )		
}		
for (i=0;i<FormOffset;I++){		
FORM_DESCRIPTOR( )		
}		
}		

The ID field is used to uniquely identify the page, and is thus used to verify hyperlinks. The Type field contains the page type, as set forth in the following table:

Type	Type code	Delay	Description
Reload	0xFF		Page should be displayed and updated continuously.
Hold	0x00	x	Page should be displayed and not updated continuously until the specified delay has passed.
AutoLink	0x01	x	Page should be displayed and then the link specified by the AUTOLINK element should be executed after the specified delay.

Referring back to the meta-data table, following the Type field and a reserved field, a Delay field is provided which contains a delay measured in seconds (for use with the types IHold and Autolink). A LinkOffset field indicates the offset of the base of the link table from the start of the meta-data. Similarly, a FocusOffset field indicates the offset of the base of the focus table from the start of the meta-data, and a FormOffset field indicates the offset of the base of the form table from the start of the meta-data.

A page element table is also provided in the meta-data, and comprises a table composed of one or more ELEMENT_DESCRIPTORs. There is an ELEMENT_DESCRIPTOR for every element (e.g., link) on the page. Each element requires additional information based on what type of element it is, and corresponds to some HTML syntax in the source. Some simple rules have been established regarding the structure of the page element table, including the rule that if an "AUTOLINK" element exists, then it is to be the first element in the element array. Also, "FORM" elements are to be located in the first entries of the page element table, after any AUTOLINK element. Form elements are to immediately follow the FORM element for the enclosing form. Lastly, the first element that can receive focus (as specified in the page element table) will receive the initial focus when the page is displayed.

The following table describes the content of an ELEMENT_DESCRIPTOR.

Syntax	No. of bits	Mnemonic
ELEMENT DESCRIPTOR( ){		
Type	8	Uimsbf
Reserved	8	Uimsbf
LinkOffset	16	Uimsbf
FocusOffset	16	Uimsbf
FormOffset	16	Uimsbf
}		

The "Type" field indicates the type of element, having one of the values set forth in the table below. The "Type" field is followed by a reserved field. A LinkOffset field indicates the offset in bytes, within the link table, of the LINK_DESCRIPTOR for this element. Similarly, a FocusOffset field indicates the offset in bytes, within the focus table, of the FOCUS_DESCRIPTOR for this element. Likewise, the FormOffset field indicates the offset in bytes, within the form table, of the FORM_DESCRIPTOR for this element. As can be appreciated, the size of an ELEMENT_DESCRIPTOR is fixed regardless of the type of the element.

The following table shows the identifying type code and the additional information that is required for each type of element, denoted by an "X":

Element Type	Type code	Focus Specifier	Link Specifier	Form Specifier
FORM	0x00		X	X
AUTOFORM	0x04		X	X
GUIDEFORM	0x08		X	X
INPUT_IMAGE	0x0C	X		X
INPUT_HIDDEN	0x10			X
INPUT_SUBMIT	0x14	X		X
INPUT_RESET	0x18	X		X
INPUT_CHECK	0x1C	X		X
INPUT_RADIO	0x20	X		X
INPUT_TEXT	0x24	X		X
INPUT_PASSWORD	0x28	X		X
AUTOLINK	0x2C		X	
LINK	0x30	X	X	

To accomplish hyperlinking, the link table contains a LINK_DESCRIPTOR for each page element that is capable of having a hyperlink, as set forth in the following table:

Syntax	No. of bits	Mnemonic
LINK_DESCRIPTOR( ){		
ID	16	Uimsbf
ProgramNumber	16	Uimsbf
ProgramElement	8	Uimsbf
PackageNumber	8	Uimsbf
}		

The ID field of the link table uniquely identifies the destination page to which an element is linked, and is used to verify hyperlinks. The ProgramNumber field contains the program_number of the program that carries the destination page, and the ProgramElement contains the program element within the program that carries that destination page. An eight-bit reserved field is also present thereafter in the link table. The PackageNumber field represents the level of tiering access, which via an authorization operation, controls the level of service to which the subscriber is entitled. Note that the size of an LINK_DESCRIPTOR is fixed.

For handling the focus operation, the focus table contains a FOCUS_DESCRIPTOR (list) for each page element that can receive focus. The FOCUS_DESCRIPTOR indicates how to draw focus on the page element. Focus may be indicated with a rectangle, a circle or a polygon as the focus ring. The FOCUS_DESCRIPTOR also indicates to which page element the focus should be shifted for each of the direction keys, (i.e., the focus chain). Note that some elements such as hidden text do not receive focus and are not part of the focus descriptor list. The following table describes the content of a FOCUS_DESCRIPTOR:

Syntax	No. of bits	Mnemonic
FOCUS_DESCRIPTOR( ){		
ElementUp	16	Uimsbf
ElementDown	16	Uimsbf
ElementRight	16	Uimsbf
ElementLeft	16	Uimsbf
Shape	8	Uimsbf
CoordCount	8	Uimsbf
For (i=0;i<CoordCount;i++){		
COORD_DESCRIPTOR( )		
}		
}		

The ElementUp field contains the index within the element table of the element that should next receive focus when the user selects the UP button. For example, in FIG. **6**, if focus is on the "SPORTS REPORT" link $\mathbf{110}_2$, a logical change in focus for the "UP" button would be to the "TO NEXT PAGE" link $\mathbf{110}_1$. Thus, the ElementUp field of the focus table for the "SPORTS REPORT" element would contain the index within the element table of the "TO NEXT PAGE" element. However, the ElementUp field of the "TO NEXT PAGE" element may be "TO NEXT PAGE," since no element is above it, or may instead toggle focus back to any of the other elements, depending on the page author's preference as influenced by agreed-upon conventions.

Similarly, the ElementDown field contains the index within the element table of the element that should receive focus next when the user selects the Down button, the ElementRight field contains the index within the element table of the element that should next receive focus when the user selects the Right button, and the ElementLeft field contains the index within the Element Table of the element that should receive focus when the user selects the Left button.

The Shape field indicates the shape of the focus ring (drawn by the OSD **96**) that will indicate focus on the given element. A 0xFF in this field indicates a polygon, a 0x00 indicates a rectangle, and a 0x01 indicates a circle. The CoordCount field contains the number of coordinates that are needed to define the focus shape, i.e., Rectangle=2, Circle=2 and Polygon=N. The following table summarizes the number and type of coordinates supplied for each type of shape:

Shape Name	Shape Code	CoordCount	Description
Polygon	0xFF	N	There are as many coordinates as there are vertices of the polygon. The COORD_DESCRIPTORS are to occur in the order that they would be encountered during a continuous tracing of the polygon edges.
Rectangle	0x00	2	The first coordinate is the upper left corner of the rectangle. The second coordinate is the lower right corner of the rectangle.
Circle	0x01	2	The first coordinate is the center of the circle. The second coordinate is the point on the radius of the circle with the minimum X value.

The format of the COORD_DESCRIPTOR is described in the following table:

Syntax	No. of bits	Mnemonic
COORD_DESCRIPTOR( ){		
X	16	Uimsbf
Y	16	Uimsbf
}		

The X field contains the X coordinate of a point on the OSD **96**, wherein the left-most pixel has an X value of zero and the right-most pixel has an X value of **351**. The Y field contains the Y coordinate of a point on the OSD **96** wherein the top-most pixel has an Y value of zero and the bottom-most pixel has a Y value of **479**.

The form table contains a FORM_DESCRIPTOR for each form element. The FORM_DESCRIPTOR contains the NAME/VALUE pair for <INPUT> tags and the ACTION URL for <FORM> tags. Note that the FORM_DESCRIPTOR has a different format for forms that access the Programming Guide, described below. The following table describes the content of a FORM_DESCRIPTOR:

Syntax	No. of bits	Mnemonic
FORM_DESCRIPTOR( ){		
if((ElementType == INPUT_SUBMIT) &		
(ElementType == INPUT_RESET) &		
(ElementType == INPUT_CHECK) &		
(ElementType == INPUT_RADIO) &		
(ElementType == INPUT_TEXT) &		
(ElementType == INPUT_PASSWORD)) {		
COORD_DESCRIPTOR( )		
COORD_DESCRIPTOR( )		
}		
if (Element is part of a GUIDEFORM){		
Function	8	Uimsbf
Reserved	8	Uimsbf
if(Function == TUNE){		
SourceID	16	Uimsbf
Time	32	Uimsbf
} else {		
COORD_DESCRIPTOR( )		
COORD_DESCRIPTOR( )		
}		
} else {		
NameLength	8	Uimsbf
ValueLength	8	Uimsbf

-continued

Syntax	No. of bits	Mnemonic
for (i=0;i<NameLength;i++){		
Name[i]	8	Uimsbf
}		
for (i=0;Value[i];i++){		
Value[i]	8	Uimsbf
}		
}		
}		

The COORD_DESCRIPTORS carry the geometry necessary to support certain form <INPUT> elements. The interpretation of these fields depends upon the type of the form element, as set forth in the following table, which shows the interpretation of these COORD_DESCRIPTORS for each type of form input element:

Element Type	Coord1	Coord2
INPUT_SUBMIT	Button Upper Left	Button Lower Right
INPUT_CHECK	Check Upper Left	
INPUT_RADIO	Check Upper Left	
INPUT_TEXT	TextBox Upper Left	TextBox Lower Right
INPUT_PASSWORD	TextBox Upper Left	TextBox Lower Right
INPUT_RESET	Button Upper Left	Button Lower Right

The Function field contains the function code for the programming guide function referenced by the GUIDEFORM. The following table shows the encoding used for each of the guide functions:

Function	Function Code
TUNE	0
REC	
HELO	2
QUIT	3
DATE	4
FROM	5
SUBJ	6
BODY	7
ACKS	8
ACKD	9
ACKB	10

Following a reserved field, the SourceID field contains a unique identifier of a program. The time field contains a Time value, wherein a zero indicates that the time should be treated as the current time.

The COORD_DESCRIPTORS for guide form elements are used to indicate where text should be drawn. The first coordinate specifies the upper left pixel of a region. The second coordinate specifies the lower right pixel of a region. The COORD_DESCRIPTORS are valid for the DATE, FROM, SUBJ and BODY guide functions.

The NameLength field contains the length of a Name string in ASCII characters, wherein a length of zero indicates that the string is not present. The ValueLength field contains the length of the Value string in ASCII characters, wherein a length of zero indicates that the string is not present. The Name[ ] field contains the ASCII Name string, if present. For form <INPUT> elements, the Name string contains the string specified by the NAME attribute. For the <FORM> element, the Name string contains the URL string specified by the ACTION attribute.

Similarly, the Value[ ] field contains the ASCII Value string. For form <INPUT> elements, the Name string contains the string specified by the VALUE attribute. For the <FORM> element, the Value string contains the method string specified by the METHOD attribute.

The following table sets forth the use of the fields within the FORM_DESCRIPTOR for other types of form elements:

Element Type	Coord1	Coord2	Name	Value
FORM			Rvalue(ACTION=)	Rvalue(METHOD=)
AUTOFORM			Rvalue(ACTION=)	Rvalue(METHOD=)
INPUT_IMAGE			Rvalue(NAME=)	Rvalue(VALUE=)
INPUT_HIDDEN			Rvalue(NAME=)	Rvalue(VALUE=)
INPUT_SUBMIT	ButtonUL	ButtonLR	Rvalue(NAME=)	Rvalue(VALUE=)
INPUT_CHECK	CheckUL		Rvalue(NAME=)	Rvalue(VALUE=)
INPUT_RADIO	CheckUL		Rvalue(NAME=)	Rvalue(VALUE=)
INPUT_TEXT	TextBoxUL	TextBoxLR	Rvalue(NAME=)	
INPUT_PASSWORD	TextBoxUL	TextBoxLR	Rvalue(NAME=)	
INPUT_RESET	ButtonUL	ButtonLR		

VI. Page Authoring

Turning to a consideration of how the page images are generated, FIG. **12** shows the general flow of one such image from authoring to downloading. First, an author creates a page using standard HTML content creation tools, such as provided in an HTML-ready authoring program **150**. Such authoring programs using standard content creation tools are well known and will not be described in detail herein. However, because the pages are to be viewed on a television, the author should take care to create an HTML page suitable for television viewing. For example, the author should select font styles and sizes appropriate for television display, and avoid color and luminance transitions which generate dot-crawl. Indeed, the author may wish to view a close approximation of the ultimately resulting page by using a VGA to NTSC converter with underscan and antiflicker filtering

enabled. In any event, as shown in FIG. **12**, an HTML source **152** including an HTML document and associated image files is provided.

After the author is satisfied with the initial result, the author runs a Phase **1** post-processing tool **154**, which performs a number of operations on the HTML source **152**. Such a tool includes the operations of rendering the page at a 640 by 480 resolution and vertically scaling the page such that the full 640 by 480 image resides within the SMPTE safe action area of a 704 by 480 overscanned television raster (approximately 640 by 432). The scaled page image is bordered by black. In addition, the rendered page is vertically filtered to reduce interlace flicker, and horizontally filtered to reduce cross-luma and chross-chroma interference artifacts.

In phase one processing, the meta-data that define how focus should be drawn on links and form elements are inferred by the HTML layout. The focus geometry is specified on a 352 by 480 grid that overlays the 704 by 480 page image. In other words, only even pixel addresses are allowed in focus specifications. The meta-data that contain the name of the HTML pages associated with each link are inferred from the HTML source **152**, as are meta-data that describe any forms. The meta-data defining the focus geometry and link URLs are encoded using standard client-side image map tags, i.e., <AREA>. The HREF attribute for entries relating to form element focus will contain the URL specified by the ACTION attribute in the <FORM> tag. Appended to that URL will be the form query string fragment defined by that form element. The HREF attribute for hyperlinks will contain the URL from the corresponding anchor or imagemap.

The output **156** of this first phase includes a true color bitmap (.BMP) file containing the processed page image and an HTML file that contains the original HTML source encapsulated in a <META> tag. Also contained are the original image files encapsulated in a <META> tag, a client-side imagemap that is the processed page image, wherein the name of the image file is identical to the name of the HTML source file, and a <MAP> tag defining the geometry of focus for all page elements that can receive focus including form elements. The HTML file also contains the meta-data for the page encapsulated in a <META> tag, and the meta-data for each page element encapsulated in a <META> tag.

Since all focus geometry is specified using standard client-side image map syntax, the author can use standard imagemap editing tools to change the focus geometry if desired. Moreover, since the page image is output as a true color .BMP file, the author can make any necessary changes to the image using standard image editing tools. Note that if desired, the anti-flicker, cross-luma, and cross-chroma filtering may be postponed to the next phase (described below).

The order of the <AREA> tags in the <MAP> tag determines the focus chain. As a result, it is possible to edit the Focus Chain for a page at this stage in order to achieve results that cannot be generated simply in the HTML source **152**. For example, the author may change the focus chain order by changing the order in which these tags occur, since the first <AREA> tag defines where focus is initially drawn when the page is displayed.

After the author is satisfied with the focus geometry, focus chain order and the final page image, a phase**2** post-processing tool **158** is run. The phase**2** tool **158** performs a number of operations, including coding the page image as a single MPEG I-Frame, and processing the <MAP> tag to generate focus descriptors for each page element. The focus descriptors for each page element are combined with the other meta-data for each page element.

The output of this second phase is a single HTML file **160** containing a number of items encapsulated in a <META> tag, including the original HTML source, the original image files, the processed page image and the compressed page image. Also contained in the file **160** are the meta-data for the page and the meta-data for each page element therein. After phase**2** processing, the page is in the intermediate page format (described above), and the author provides the final file to the carousel building tool **146**, such as by running a file transfer program **162** to transfer the file via the Internet **48** to the server **46** of the Information Service.

Note that the phase1 and phase**2** processes (**154** and **158**, respectively) can be run back-to-back without intervention. The break exists therebetween to provide authors the ability to exercise more control over the process if so desired. Thus, as described above, between phase1 **154** and phase**2** **158** of the tool chain, an author may edit the focus geometry by using any number of existing imagemap editing tools, while the focus chain may also be edited by editing the FOCUS meta-data.

To allow the HTML author to reference user-specific data stored in the set-top-box **28** and perform other limited functions, a set of form element names are defined to have special meanings, as set forth in the table below:

Element name	Meaning and usage	Format
_SHIPNAME	Default shipping name. May be used for transactions and bingo card.	
_SHIPSTREET		
_SHIPCITY		
_SHIPSTATE		
_SHIPZIP		
_CARDNAME	Default cardholder name. May be used for one-button purchase.	
_CARDTYPE		
CARDMONTH		
_CARDYEAR		
_EMAIL	Default Email address for this box.	Standard RFC822 address syntax.
_TIME	Current time.	
_QUANTITY	Name to use for quantity input on transaction form in order to enable _TOTAL.	
_PRICE	Name to use for price hidden text on transaction form in order to enable _TOTAL	

-continued

Element name	Meaning and usage	Format
_SANDH	Name to use for shipping and handling etc hidden text on transaction form if needed for _TOTAL	
_TOTAL	Causes the Browser 62 to evaluate the following expression that can be placed in a read-only text entry element. _TOTAL = _QUANTITY * _PRICE + _SANDH	

When included in the body of a form as hidden text, these elements generate a standard name/value pair in the form query string. For example, the tag <INPUT TYPE= HIDDEN NAME="_EMAIL"> will generate the following sub-string in the form query string: "&_EMAIL=set_top_box_name@customers.tci.com"

VII. Other

At least some existing programming guide applications and cable system infrastructures support one-way e-mail broadcasts to all users. This function can be extended to allow one-way e-mail to a specific subscriber, using the existing (e.g. PreVue) e-mail notification and display mechanism. E-mail data can be sent to a specific set-top box via the out-of-band modulator **60** (FIG. **2**) and the out-of-band tuner **71** (FIG. **3**), in the same manner that a specific box is enabled for receiving a pay-per-view movie. Although not directly linked to the Information Service architecture, a head-end e-mail server is considered a part of the Information Service head-end installation, and the e-mail server software is supplied by the assignee of the present invention.

The Information Service e-mail client is implemented using standard HTML forms. A simple example would be a form containing buttons for "Previous", "Next", and "Delete" which would submit the appropriate query strings to the guide in order to have the mail text rendered using the OSD **96** over the page image. The author of the page can specify where the "From" address is rendered and where the message text is rendered. The coordinates for these fields are referenced to a 352 by 480 OSD grid, where the original 640 by 480 page image underlays the center 320 by 432 pixels.

Although at present e-mail is generally read-only, it is understood that two-way e-mail is feasible. For example, outgoing e-mail messages may be written to the memory **82** and polled via the modem **134** (FIG. **3**) by an e-mail server or the like, or sent directly (or as soon as a telephone line is available) thereto via the modem **134** upon user command. Moreover, a backchannel which sends information on the transmission medium **24** from the various set-top boxes back to the head-end **22** may be dedicated, in which event all data, including form entry data and e-mail data, can be immediately transferred without polling the modem **134**. However, in order for e-mail to be practical, a more sophisticated alphanumeric input device, such as a conventional QWERTY keyboard arranged for communicating with the set-top box, would likely need to be provided.

It is also feasible to add audio to the carousel **50**, whereby audio may be optionally played while the user is in the Information Service. In such a system, part of the information sent with each transmitted page is the next block (or blocks, if multiple, user-selectable audio streams are transmitted) of encoded audio data. Depending on the amount of memory in the set-top box, the audio data may be buffered so as to load more than one carousel revolution's worth (e.g. eight seconds) of audio data from a given carousel **50** into the set-top box **28** for each stream. However, by regularly changing the carousel, the audio may be a real-time broadcast.

Lastly, instead of a carousel **50**, it is feasible to have the server **46** execute a program that selectively injects the pages into the transport stream. For example, based on statistical information such as page popularity in conjunction with the time of day, (information which may be sampled and polled from the set-top box **28**), more popular pages can be injected more frequently into the stream than less popular pages. The maximum latency for a given page can thus be controlled, without effecting the total number of pages possible, (which occurs with a carousel by increasing a page's frequency in the carousel at the expense of decreasing the number of distinct pages therein).

As can be seen from the foregoing detailed description, there is provided an interactive entertainment and information system using a cable television set-top box, wherein existing digital set top boxes may be used without need for modification of the hardware therein. The system enables a user to hyperlink between pages of information, and facilitates an interactive relationship between transmitted programming and information pages related thereto. The user may submit form information, including forms related to a viewed television program. The system provides a simple to operate, low cost information service, while being flexible and extensible.

While the invention is susceptible to various modifications and alternative constructions, a certain illustrated embodiment thereof is shown in the drawings and has been described above in detail. It should be understood, however, that there is no intention to limit the invention to the specific form disclosed, but on the contrary, the intention is to cover all modifications, alternative constructions, and equivalents falling within the spirit and scope of the invention.

What is claimed is:

1. An interactive television system, comprising,

a head-end having means for injecting video information into a transmission medium, the video information including a plurality of page images, the video information further including meta-data associated with at least one of the plurality of page images, the meta-data for the at least one page image being transmitted with the page image;

means at the subscriber end for detecting a first request to display one of the plurality of page images;

means for selecting and outputting video signals representing the page image upon detection of the first request, the page image including at least one active location having an action corresponding thereto;

means at the subscriber end for detecting a second request to take the action corresponding to the active location; and

means for taking the action upon detection of the second request, including means for submitting a form.

2. The system of claim **1** wherein the means for taking the action includes means for reading the meta-data in the video information.

3. The system of claim 2 wherein the means for taking action further includes means for recognizing a link to another page and for causing video signals representing said page to be output.

4. The system of claim 2 wherein the means for taking action includes means for writing user-data to a memory at the subscriber end.

5. The system of claim 1 wherein the form includes purchasing information.

6. The system of claim 5 further comprising means at the subscriber end and the head-end for communicating user-data from the memory to the head-end.

7. The system of claim 5 wherein the means for taking action includes means for changing the output video signal to reflect the state of the user-data.

8. The system of claim 7 wherein the means for changing the output vide o signal includes an on-screen display at the subscriber end.

9. The system of claim 1 wherein the means for detecting the first and second requests include a user-input device linked to a set-top box.

10. The system of claim 1 wherein the second request corresponds to a directional command, and the means for taking action include means for changing the active location in response to the directional command.

11. The system of claim 1 wherein the video information is arranged as a carousel of video images such that the images are periodically injected into the transmission medium.

12. The system of claim 11 further comprising means for rebuilding the carousel with a change to at least one page image in the plurality of page images.

13. The system of claim 12 wherein the carousel is rebuilt a plurality of times such that a page image appears to have animated motion thereon.

14. The system of claim 1 wherein the injected video information includes electronic mail.

15. The interactive television system of claim 1, wherein the meta-data and corresponding page image are transmitted as a single transport stream.

16. The interactive television system of claim 15, wherein the single transport stream comprises an MPEG2 transport stream.

17. In a television-based system having a head-end for providing video information to set-top boxes of users, a method of enabling users to interact with said video information, comprising:

injecting video information into a transport stream of a transmission medium for transmission to the set-top boxes, the video information including a plurality of page images and meta-data associated with the page images;

detecting at the set-top box a first user request to display one of the plurality of page images;

selecting a page image based on the user request;

outputting video signals representing the selected page image from the set-top box, the meta-data associated with the page image identifying at least one active location and an action corresponding thereto;

detecting at the set-top box a second request to take the action corresponding to the active location;

taking the action including writing user-data to a memory at the subscriber end; and

communicating user-data from the memory to the head-end.

18. The method of claim 17 wherein the step of taking action includes the steps of recognizing a link to another page in the meta-data and causing video signals representing said page to be output.

19. The method of claim 17 wherein the step of taking action includes the step of submitting a form.

20. The method of claim 17 wherein the step of taking an action includes the steps of receiving user-data and changing the output video signal to reflect the state of the user-data.

21. The method of claim 20 wherein the step of changing the output video signal includes the step of controlling an on-screen display in the set-top box.

22. The method of claim 17 wherein the steps of detecting the first request and the second requests include the step of receiving commands from a user-input device.

23. The method of claim 17 wherein the step of injecting the page images includes the step of periodically repeating the injection of page images.

24. The method of claim 17 wherein the step of receiving a second request includes the step of receiving a directional command, and the step of taking action includes the step of changing the active location on the page image in response to the directional command.

25. The method of claim 17 wherein the step of injecting a plurality of page images includes the steps of changing at least one page image in the plurality of page images, and periodically repeating the injection of page images.

26. The method of claim 25 wherein a page image is changed a plurality of times such that a page image appears to have animated motion thereon.

27. The method of claim 17 wherein the injected video information includes electronic mail, and outputting video signals representing the selected page image includes outputting electronic mail information.

28. The method of claim 17, wherein the transport stream comprises a single broadcast channel program, and wherein at least one of the page images and the meta-data associated with the at least one page images is transmitted over the single broadcast channel.

29. The method of claim 28, wherein the at least one page image is transmitted as a program element of the broadcast channel program, and wherein the meta-data associated with the at least one page image is transmitted as another program element of the same broadcast channel program.

30. An interactive television system, comprising:

a head-end having means for periodically injecting video information into a transmission medium, the video information including a plurality of page images, the plurality of page images being arranged as a carousel of video images;

means for rebuilding the carousel a plurality of times with a change to at least one video image in the plurality of page images such that a page image appears to have animated motion;

means at the subscriber end for detecting a first request to display one of the plurality of page images;

means for selecting and outputting video signals representing the page image upon detection of the first request, the page image including at least one active location having a corresponding action;

means at the subscriber end for detecting a second request to take the action corresponding to the active location; and

means for taking the action upon detection of the second request.

31. In a television-based system having a head-end for providing video information to set-top boxes, a method of enabling users to interact with said video information, comprising:

injecting a plurality of page images as video information into a transmission medium for transmission to the set-top boxes, the video information including meta-data associated with the page images, wherein injecting the plurality of page images includes changing at least one page image in the plurality of page images, and periodically repeating the injection of page images;

detecting at the set-top box a first user request to display one of the plurality of page images;

selecting as a selected page image a page image based on the user request;

outputting video signals representing the selected page image from the set-top box including changing the selected page image a plurality of times such that the page image appears to have animated motion thereon, the meta-data associated with the page image identifying at least one active location and a corresponding action;

detecting at the set-top box a second request to take the action corresponding to the active location; and

taking the action corresponding to the active location.

32. In a television-based system having a head-end for providing video information to set-top boxes of users, a method of enabling users to interact with said video information, comprising:

injecting a plurality of page images as video information into a transmission medium for transmission to the set-top boxes, the video information including meta-data associated with the page images;

detecting at the set-top box a first user request to display one of the plurality of page images;

selecting a page image based on the user request;

outputting video signals representing the selected page image from the set-top box, the meta-data associated with the page image identifying at least one active location and an action corresponding thereto;

detecting at the set-top box a second request to take the action corresponding to the active location;

taking the action, including submitting a form.

33. In a television-based system having a head-end for providing video information to set-top boxes of users, a method of enabling users to interact with said video information, comprising:

injecting a plurality of page images as video information into a transmission medium for transmission to the set-top boxes, the video information including meta-data associated with the page images;

detecting at the set-top box a first user request to display one of the plurality of page images;

selecting a page image based on the user request;

outputting video signals representing the selected page image from the set-top box, the meta-data associated with the page image identifying at least one active location and an action corresponding thereto;

detecting at the set-top box a second request to take the action corresponding to the active location; and

taking the action including receiving user-data and changing the output video signal to reflect the state of the user-data.

34. The method of claim **33** wherein changing the output video signal includes controlling an on-screen display in the set-top box.

* * * * *

1 of 32 DOCUMENTS

Caution
As of: Jun 20, 2007

KSR INTERNATIONAL CO., PETITIONER v. TELEFLEX INC. ET AL.

No. 04-1350

SUPREME COURT OF THE UNITED STATES

127 S. Ct. 1727; 167 L. Ed. 2d 705; 2007 U.S. LEXIS 4745; 75 U.S.L.W. 4289; 82 U.S.P.Q.2D (BNA) 1385; 20 Fla. L. Weekly Fed. S 248

November 28, 2006, Argued
April 30, 2007, Decided

NOTICE: [***1] The LEXIS pagination of this document is subject to change pending release of the final published version.

PRIOR HISTORY: ON WRIT OF CERTIORARI TO THE UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT. *Teleflex, Inc. v. KSR Int'l Co., 119 Fed. Appx. 282, 2005 U.S. App. LEXIS 176 (Fed. Cir., 2005)*

DISPOSITION: Reversed and remanded.

Case in Brief ($)

Expert Commentary ($)

Donald S. Chisum on the Supreme Court's Latest Word on Obviousness and Combination Inventions

Common sense governs when assessing the patentability of inventions that combine prior art elements. So the Supreme Court holds in KSR. It affirms its prior decisions, which urged caution in granting patents on combinations. It rejects rigid applications of the Federal Circuit's "TSM" (teaching, suggestion, motivation) test, which made challenges to combination patents more difficult. This commentary, written by Donald S. Chisum, author of the renowned treatise *Chisum on Patents* discusses the import and potential extensions of this important decision.

SYLLABUS: To control a conventional automobile's speed, the driver depresses or releases the gas pedal, which interacts with the throttle via a cable or other mechanical link. Because the pedal's position in the footwell normally cannot be adjusted, a driver wishing to be closer or farther from it must either reposition himself in the seat or move the seat, both of which can be imperfect solutions for smaller drivers in cars with deep footwells. This prompted inventors to design and patent pedals that could be adjusted to change their locations. The Asano patent reveals a support structure whereby, when the pedal location is [***2] adjusted, one of the pedal's pivot points stays fixed. Asano is also designed so that the force necessary to depress the pedal is the same regardless of location adjustments. The Redding patent reveals a different, sliding mechanism where both the pedal and the pivot point are adjusted.

In newer cars, computer-controlled throttles do not operate through force transferred from the pedal by a mechanical link, but open and close valves in response to electronic signals. For the computer to know what is happening with the pedal, an electronic sensor must translate the mechanical operation into digital data. Inventors had obtained a number of patents for such sensors. The so-called *'936 patent* taught that it was preferable to detect the pedal's position in the pedal mechanism, not in the engine, so the patent disclosed a pedal with an electronic sensor on a pivot point in the pedal assembly. The Smith patent taught that to prevent the wires connecting the sensor to the computer from chafing and wearing out, the sensor should be put on a fixed part of the pedal assembly rather than in or on the pedal's footpad. Inventors had also patented self-contained modular sensors, which can be [***3] taken off the shelf and attached to any mechanical pedal to allow it to function with a computer-controlled throttle. The *'068 patent*

disclosed one such sensor. Chevrolet also manufactured trucks using modular sensors attached to the pedal support bracket, adjacent to the pedal and engaged with the pivot shaft about which the pedal rotates. Other patents disclose electronic sensors attached to adjustable pedal assemblies. For example, the Rixon patent locates the sensor in the pedal footpad, but is known for wire chafing.

After petitioner KSR developed an adjustable pedal system for cars with cable-actuated throttles and obtained its *'976 patent* for the design, General Motors Corporation (GMC) chose KSR to supply adjustable pedal systems for trucks using computer-controlled throttles. To make the *'976* pedal compatible with the trucks, KSR added a modular sensor to its design. Respondents (Teleflex) hold the exclusive license for the Engelgau patent, claim 4 of which discloses a position-adjustable pedal assembly with an electronic pedal position sensor attached a fixed pivot point. Despite having denied a similar, broader claim, the U.S. Patent and Trademark Office (PTO) had allowed [***4] claim 4 because it included the limitation of a fixed pivot position, which distinguished the design from Redding's. Asano was neither included among the Engelgau patent's prior art references nor mentioned in the patent's prosecution, and the PTO did not have before it an adjustable pedal with a fixed pivot point. After learning of KSR's design for GMC, Teleflex sued for infringement, asserting that KSR's pedal system infringed the Engelgau patent's claim 4. KSR countered that claim 4 was invalid under *§ 103* of the Patent Act, which forbids issuance of a patent when "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art."

Graham v. John Deere Co. of Kansas City, 383 U.S. 1, 17-18, 86 S. Ct. 684, 15 L. Ed. 2d 545, set out an objective analysis for applying *§ 103*: "The scope and content of the prior art are . . . determined; differences between the prior art and the claims at issue are . . . ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background the obviousness or nonobviousness [***5] of the subject matter is determined. Such secondary considerations as commercial success, long felt but unsolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented." While the sequence of these questions might be reordered in any particular case, the factors define the controlling inquiry. However, seeking to resolve the obviousness question with more uniformity and consistency, the Federal Circuit has employed a "teaching, suggestion, or motivation" (TSM) test, under which a patent claim is only proved obvious if the prior art, the problem's nature, or the knowledge of a person having ordinary skill in the art reveals some motivation or suggestion to combine the prior art teachings.

The District Court granted KSR summary judgment. After reviewing pedal design history, the Engelgau patent's scope, and the relevant prior art, the court considered claim 4's validity, applying *Graham*'s framework to determine whether under summary-judgment standards KSR had demonstrated that claim 4 was obvious. The court found "little difference" between the prior art's teachings and claim 4: [***6] Asano taught everything contained in the claim except using a sensor to detect the pedal's position and transmit it to a computer controlling the throttle. That additional aspect was revealed in, *e.g.*, the *'068 patent* and Chevrolet's sensors. The court then held that KSR satisfied the TSM test, reasoning (1) the state of the industry would lead inevitably to combinations of electronic sensors and adjustable pedals, (2) Rixon provided the basis for these developments, and (3) Smith taught a solution to Rixon's chafing problems by positioning the sensor on the pedal's fixed structure, which could lead to the combination of a pedal like Asano with a pedal position sensor.

Reversing, the Federal Circuit ruled the District Court had not applied the TSM test strictly enough, having failed to make findings as to the specific understanding or principle within a skilled artisan's knowledge that would have motivated one with no knowledge of the invention to attach an electronic control to the Asano assembly's support bracket. The Court of Appeals held that the District Court's recourse to the nature of the problem to be solved was insufficient because, unless the prior art references [***7] addressed the precise problem that the patentee was trying to solve, the problem would not motivate an inventor to look at those references. The appeals court found that the Asano pedal was designed to ensure that the force required to depress the pedal is the same no matter how the pedal is adjusted, whereas Engelgau sought to provide a simpler, smaller, cheaper adjustable electronic pedal. The Rixon pedal, said the court, suffered from chafing but was not designed to solve that problem and taught nothing helpful to Engelgau's purpose. Smith, in turn, did not relate to adjustable pedals and did not necessarily go to the issue of motivation to attach the electronic control on the pedal assembly's support bracket. So interpreted, the court held, the patents would not have led a person of ordinary skill to put a sensor on an Asano-like pedal. That it might have been obvious to try that combination was likewise irrelevant. Finally, the court held that genuine issues of material fact precluded summary judgment.

Held: The Federal Circuit addressed the obviousness question in a narrow, rigid manner that is inconsistent with *§ 103* and this Court's precedents. KSR provided convincing [***8] evidence that mounting an available sensor on a fixed pivot point of the Asano pedal was a design step well within the grasp of a person of ordinary skill in the relevant art and that the benefit of doing so would be obvious. Its arguments, and the record, demonstrate that the Engelgau patent's claim 4 is obvious. Pp. 11-24.

1. *Graham* provided an expansive and flexible approach to the obviousness question that is inconsistent with the way the Federal Circuit applied its TSM test here. Neither *§ 103*'s enactment nor *Graham*'s analysis disturbed the Court's earlier instructions concerning the need for caution in granting a patent based on the combination of elements found in the prior art. See *Great Atlantic & Pacific Tea Co. v. Supermarket Equipment Corp., 340 U.S. 147, 152, 71 S. Ct. 127, 95 L. Ed. 162, 1951 Dec. Comm'r Pat. 572.* Such a combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. See, *e.g., United States v. Adams, 383 U.S. 39, 50-52, 86 S. Ct. 708, 15 L. Ed. 2d 572, 174 Ct. Cl. 1293.* When a work is available in one field, design incentives and other market forces can prompt variations of it, either in the same field or in another. If a person [***9] of ordinary skill in the art can implement a predictable variation, and would see the benefit of doing so, *§ 103* likely bars its patentability. Moreover, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond that person's skill. A court must ask whether the improvement is more than the predictable use of prior-art elements according to their established functions. Following these principles may be difficult if the claimed subject matter involves more than the simple substitution of one known element for another or the mere application of a known technique to a piece of prior art ready for the improvement. To determine whether there was an apparent reason to combine the known elements in the way a patent claims, it will often be necessary to look to interrelated teachings of multiple patents; to the effects of demands known to the design community or present in the marketplace; and to the background knowledge possessed by a person having ordinary skill in the art. To facilitate review, this analysis should [***10] be made explicit. But it need not seek out precise teachings directed to the challenged claim's specific subject matter, for a court can consider the inferences and creative steps a person of ordinary skill in the art would employ. Pp. 11-14.

(b) The TSM test captures a helpful insight: A patent composed of several elements is not proved obvious merely by demonstrating that each element was, independently, known in the prior art. Although common sense directs caution as to a patent application claiming as innovation the combination of two known devices according to their established functions, it can be important to identify a reason that would have prompted a person of ordinary skill in the art to combine the elements as the new invention does. Inventions usually rely upon building blocks long since uncovered, and claimed discoveries almost necessarily will be combinations of what, in some sense, is already known. Helpful insights, however, need not become rigid and mandatory formulas. If it is so applied, the TSM test is incompatible with this Court's precedents. The diversity of inventive pursuits and of modern technology counsels against confining the obviousness analysis [***11] by a formalistic conception of the words teaching, suggestion, and motivation, or by overemphasizing the importance of published articles and the explicit content of issued patents. In many fields there may be little discussion of obvious techniques or combinations, and market demand, rather than scientific literature, may often drive design trends. Granting patent protection to advances that would occur in the ordinary course without real innovation retards progress and may, for patents combining previously known elements, deprive prior inventions of their value or utility. Since the TSM test was devised, the Federal Circuit doubtless has applied it in accord with these principles in many cases. There is no necessary inconsistency between the test and the *Graham* analysis. But a court errs where, as here, it transforms general principle into a rigid rule limiting the obviousness inquiry. Pp. 14-15.

(c) The flaws in the Federal Circuit's analysis relate mostly to its narrow conception of the obviousness inquiry consequent in its application of the TSM test. The Circuit first erred in holding that courts and patent examiners should look only to the problem the patentee was trying [***12] to solve. Under the correct analysis, any need or problem known in the field and addressed by the patent can provide a reason for combining the elements in the manner claimed. Second, the appeals court erred in assuming that a person of ordinary skill in the art attempting to solve a problem will be led only to those prior art elements designed to solve the same problem. The court wrongly concluded that because Asano's primary purpose was solving the constant ratio problem, an inventor considering how to put a sensor on an adjustable pedal would have no reason to consider putting it on the Asano pedal. It is common sense that familiar items may have obvious uses beyond their primary purposes, and a person of ordinary skill often will be able to fit the teach-

ings of multiple patents together like pieces of a puzzle. Regardless of Asano's primary purpose, it provided an obvious example of an adjustable pedal with a fixed pivot point, and the prior art was replete with patents indicating that such a point was an ideal mount for a sensor. Third, the court erred in concluding that a patent claim cannot be proved obvious merely by showing that the combination of elements was obvious to try. [***13] When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill in the art has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. Finally, the court drew the wrong conclusion from the risk of courts and patent examiners falling prey to hindsight bias. Rigid preventative rules that deny recourse to common sense are neither necessary under, nor consistent with, this Court's case law. Pp. 15-18.

2. Application of the foregoing standards demonstrates that claim 4 is obvious. Pp. 18-23.

(a) The Court rejects Teleflex's argument that the Asano pivot mechanism's design prevents its combination with a sensor in the manner claim 4 describes. This argument was not raised before the District Court, and it is unclear whether it was raised before the Federal Circuit. Given the significance of the District Court's finding that combining Asano with a pivot-mounted pedal position sensor fell within claim 4's scope, it is apparent that Teleflex would [***14] have made clearer challenges if it intended to preserve this claim. Its failure to clearly raise the argument, and the appeals court's silence on the issue, lead this Court to accept the District Court's conclusion. Pp. 18-20.

(b) The District Court correctly concluded that when Engelgau designed the claim 4 subject matter, it was obvious to a person of ordinary skill in the art to combine Asano with a pivot-mounted pedal position sensor. There then was a marketplace creating a strong incentive to convert mechanical pedals to electronic pedals, and the prior art taught a number of methods for doing so. The Federal Circuit considered the issue too narrowly by, in effect, asking whether a pedal designer writing on a blank slate would have chosen both Asano and a modular sensor similar to the ones used in the Chevrolet trucks and disclosed in the *'068 patent*. The proper question was whether a pedal designer of ordinary skill in the art, facing the wide range of needs created by developments in the field, would have seen an obvious benefit to upgrading Asano with a sensor. For such a designer starting with Asano, the question was where to attach the sensor. The *'936 patent* taught [***15] the utility of putting the sensor on the pedal device. Smith, in turn, explained not to put the sensor on the pedal footpad, but instead on the structure. And from Rixon's known wire-chafing problems, and Smith's teaching that the pedal assemblies must not precipitate any motion in the connecting wires, the designer would know to place the sensor on a nonmoving part of the pedal structure. The most obvious such point is a pivot point. The designer, accordingly, would follow Smith in mounting the sensor there. Just as it was possible to begin with the objective to upgrade Asano to work with a computer-controlled throttle, so too was it possible to take an adjustable electronic pedal like Rixon and seek an improvement that would avoid the wire-chafing problem. Teleflex has not shown anything in the prior art that taught away from the use of Asano, nor any secondary factors to dislodge the determination that claim 4 is obvious. Pp. 20-23.

3. The Court disagrees with the Federal Circuit's holding that genuine issues of material fact precluded summary judgment. The ultimate judgment of obviousness is a legal determination. *Graham, 383 U.S., at 17, 86 S. Ct. 684, 15 L. Ed. 2d 545*. Where, as here, the [***16] prior art's content, the patent claim's scope, and the level of ordinary skill in the art are not in material dispute and the claim's obviousness is apparent, summary judgment is appropriate. P. 23.

119 Fed. Appx. 282, reversed and remanded.

COUNSEL: James W. Dabneyargued the cause for petitioner.

Thomas G. Hungarargued the cause for the United States, as amicus curiae, by special leave of court.

Thomas C. Goldstein

JUDGES: KENNEDY, J., delivered the opinion for a unanimous Court.

OPINION BY: KENNEDY

OPINION:

[**714] [*1734] JUSTICE KENNEDY delivered the opinion of the Court.

Teleflex Incorporated and its subsidiary Technology Holding Company -- both referred to here as Teleflex -- sued KSR International Company for patent infringement. The patent at issue, *United States Patent No. 6,237,565* B1, is entitled "Adjustable Pedal Assembly With Electronic Throttle Control." Supplemental App. 1.

The patentee is Steven J. Engelgau, and the patent is referred to as "the Engelgau patent." Teleflex holds the exclusive license to the patent.

Claim 4 of the Engelgau patent describes a mechanism for combining an electronic sensor with an adjustable automobile pedal so the pedal's position can be transmitted to a computer that controls the throttle in the vehicle's engine. When Teleflex accused KSR of infringing the Engelgau patent by adding an electronic sensor to one of KSR's previously [***17] designed pedals, KSR countered that claim 4 was invalid under the Patent Act, *35 U.S.C. § 103*, because its subject matter was obvious.

Section 103 forbids issuance of a patent when "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having [**715] ordinary skill in the art to which said subject matter pertains."

In *Graham v. John Deere Co. of Kansas City, 383 U.S. 1, 86 S. Ct. 684, 15 L. Ed. 2d 545 (1966)*, the Court set out a framework for applying the statutory language of *§ 103*, language itself based on the logic of the earlier decision in *Hotchkiss v. Greenwood, 52 U.S. 248, 11 How. 248, 13 L. Ed. 683 (1851)*, and its progeny. See *383 U.S., at 15-17, 86 S. Ct. 684, 15 L. Ed. 2d 545*. The analysis is objective:

> "Under *§ 103*, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background the obviousness or nonobviousness of the subject matter is determined. Such secondary considerations [***18] as commercial success, long felt but unsolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented." *Id., at 17-18, 86 S. Ct. 684, 15 L. Ed. 2d 545.*

While the sequence of these questions might be reordered in any particular case, the factors continue to define the inquiry that controls. If a court, or patent examiner, conducts this analysis and concludes the claimed subject matter was obvious, the claim is invalid under *§ 103*.

Seeking to resolve the question of obviousness with more uniformity and consistency, the Court of Appeals for the Federal Circuit has employed an approach referred to by the parties as the "teaching, suggestion, or motivation" test (TSM test), under which a patent claim is only proved obvious if "some motivation or suggestion to combine the prior art teachings" can be found in the prior art, the nature of the problem, or the knowledge of a person having ordinary skill in the art. See, *e.g., Al-Site Corp. v. VSI Int'l, Inc., 174 F.3d 1308, 1323-1324 (CA Fed. 1999)*. KSR challenges that [*1735] test, or at least its application in this case. See *119 Fed. Appx. 282, 286-290 (CA Fed. 2005)*. [***19] Because the Court of Appeals addressed the question of obviousness in a manner contrary to *§ 103* and our precedents, we granted certiorari, *547 U.S. , 126 S. Ct. 2965, 165 L. Ed. 2d 949 (2006)*. We now reverse.

I

A

In car engines without computer-controlled throttles, the accelerator pedal interacts with the throttle via cable or other mechanical link. The pedal arm acts as a lever rotating around a pivot point. In a cable-actuated throttle control the rotation caused by pushing down the pedal pulls a cable, which in turn pulls open valves in the carburetor or fuel injection unit. The wider the valves open, the more fuel and air are released, causing combustion to increase and the car to accelerate. When the driver takes his foot off the pedal, the opposite occurs as the cable is released and the valves slide closed.

In the 1990's it became more common to install computers in cars to control engine operation. Computer-controlled throttles open and close valves in response to electronic signals, not through force transferred from the pedal by a mechanical link. Constant, delicate adjustments of air and fuel mixture are possible. The computer's rapid processing of factors beyond the pedal's position improves [***20] [**716] fuel efficiency and engine performance.

For a computer-controlled throttle to respond to a driver's operation of the car, the computer must know what is happening with the pedal. A cable or mechanical link does not suffice for this purpose; at some point, an electronic sensor is necessary to translate the mechanical operation into digital data the computer can understand.

Before discussing sensors further we turn to the mechanical design of the pedal itself. In the traditional design a pedal can be pushed down or released but cannot have its position in the footwell adjusted by sliding the pedal forward or back. As a result, a driver who wishes to be closer or farther from the pedal must either reposition himself in the driver's seat or move the seat in some way. In cars with deep footwells these are imperfect solutions for drivers of smaller stature. To solve the prob-

lem, inventors, beginning in the 1970's, designed pedals that could be adjusted to change their location in the footwell. Important for this case are two adjustable pedals disclosed in *U.S. Patent Nos. 5,010,782* (filed July 28, 1989) (Asano) and *5,460,061* (filed Sept. 17, 1993) (Redding). The Asano patent reveals a [***21] support structure that houses the pedal so that even when the pedal location is adjusted relative to the driver, one of the pedal's pivot points stays fixed. The pedal is also designed so that the force necessary to push the pedal down is the same regardless of adjustments to its location. The Redding patent reveals a different, sliding mechanism where both the pedal and the pivot point are adjusted.

We return to sensors. Well before Engelgau applied for his challenged patent, some inventors had obtained patents involving electronic pedal sensors for computer-controlled throttles. These inventions, such as the device disclosed in *U.S. Patent No. 5,241,936* (filed Sept. 9, 1991) ('936), taught that it was preferable to detect the pedal's position in the pedal assembly, not in the engine. The *'936 patent* disclosed a pedal with an electronic sensor on a pivot point in the pedal assembly. *U.S. Patent No. 5,063,811* (filed July 9, 1990) (Smith) taught that to prevent the [*1736] wires connecting the sensor to the computer from chafing and wearing out, and to avoid grime and damage from the driver's foot, the sensor should be put on a fixed part of the pedal assembly rather than in or on the pedal's [***22] footpad.

In addition to patents for pedals with integrated sensors inventors obtained patents for self-contained modular sensors. A modular sensor is designed independently of a given pedal so that it can be taken off the shelf and attached to mechanical pedals of various sorts, enabling the pedals to be used in automobiles with computer-controlled throttles. One such sensor was disclosed in *U.S. Patent No. 5,385,068* (filed Dec. 18, 1992) ('068). In 1994, Chevrolet manufactured a line of trucks using modular sensors "attached to the pedal support bracket, adjacent to the pedal and engaged with the pivot shaft about which the pedal rotates in operation." *298 F. Supp. 2d 581, 589 (ED Mich. 2003).*

The prior art contained patents involving the placement of sensors on adjustable pedals as well. For example, *U.S. Patent No. 5,819,593* (filed Aug. 17, 1995) (Rixon) discloses an adjustable pedal assembly with an [**717] electronic sensor for detecting the pedal's position. In the Rixon pedal the sensor is located in the pedal footpad. The Rixon pedal was known to suffer from wire chafing when the pedal was depressed and released.

This short account of pedal and sensor technology leads [***23] to the instant case.

B

KSR, a Canadian company, manufactures and supplies auto parts, including pedal systems. Ford Motor Company hired KSR in 1998 to supply an adjustable pedal system for various lines of automobiles with cable-actuated throttle controls. KSR developed an adjustable mechanical pedal for Ford and obtained *U.S. Patent No. 6,151,976* (filed July 16, 1999) ('976) for the design. In 2000, KSR was chosen by General Motors Corporation (GMC or GM) to supply adjustable pedal systems for Chevrolet and GMC light trucks that used engines with computer-controlled throttles. To make the *'976* pedal compatible with the trucks, KSR merely took that design and added a modular sensor.

Teleflex is a rival to KSR in the design and manufacture of adjustable pedals. As noted, it is the exclusive licensee of the Engelgau patent. Engelgau filed the patent application on August 22, 2000 as a continuation of a previous application for *U.S. Patent No. 6,109,241*, which was filed on January 26, 1999. He has sworn he invented the patent's subject matter on February 14, 1998. The Engelgau patent discloses an adjustable electronic pedal described in the specification as a "simplified vehicle control [***24] pedal assembly that is less expensive, and which uses fewer parts and is easier to package within the vehicle." Engelgau, col. 2, lines 2-5, Supplemental App. 6. Claim 4 of the patent, at issue here, describes:

"A vehicle control pedal apparatus comprising:

> a support adapted to be mounted to a vehicle structure;
>
> an adjustable pedal assembly having a pedal arm moveable in fore and aft directions with respect to said support;
>
> a pivot for pivotally supporting said adjustable pedal assembly with respect to said support and defining a pivot axis; and
>
> an electronic control attached to said support for controlling a vehicle system;
>
> said apparatus characterized by said electronic control being responsive to said pivot for providing a signal that corresponds to pedal arm position as said pedal arm pivots about said pivot [*1737] axis between rest and applied positions wherein the position of said pivot remains constant while said pedal arm moves in fore and aft directions with respect to said pivot." *Id.*, col. 6, lines 17-36, Supple-

> mental App. 8 (diagram numbers omitted).

We agree with the District Court that the claim discloses "a position-adjustable pedal [***25] assembly with an electronic pedal position sensor attached to the support member of the pedal assembly. Attaching the sensor to the support member allows the sensor to remain in a fixed position while the driver adjusts the pedal." *298 F. Supp. 2d, at 586-587.*

Before issuing the Engelgau patent the U.S. Patent and Trademark Office (PTO) rejected one of the patent claims that was similar to, but [**718] broader than, the present claim 4. The claim did not include the requirement that the sensor be placed on a fixed pivot point. The PTO concluded the claim was an obvious combination of the prior art disclosed in Redding and Smith, explaining:

> "'Since the prior art references are from the field of endeavor, the purpose disclosed . . . would have been recognized in the pertinent art of Redding. Therefore it would have been obvious . . . to provide the device of Redding with the . . . means attached to a support member as taught by Smith.'" *Id., at 595.*

In other words Redding provided an example of an adjustable pedal and Smith explained how to mount a sensor on a pedal's support structure, and the rejected patent claim merely put these two teachings together. [***26]

Although the broader claim was rejected, claim 4 was later allowed because it included the limitation of a fixed pivot point, which distinguished the design from Redding's. *Ibid.* Engelgau had not included Asano among the prior art references, and Asano was not mentioned in the patent's prosecution. Thus, the PTO did not have before it an adjustable pedal with a fixed pivot point. The patent issued on May 29, 2001 and was assigned to Teleflex.

Upon learning of KSR's design for GM, Teleflex sent a warning letter informing KSR that its proposal would violate the Engelgau patent. "'Teleflex believes that any supplier of a product that combines an adjustable pedal with an electronic throttle control necessarily employs technology covered by one or more'" of Teleflex's patents. *Id., at 585.* KSR refused to enter a royalty arrangement with Teleflex; so Teleflex sued for infringement, asserting KSR's pedal infringed the Engelgau patent and two other patents. *Ibid.* Teleflex later abandoned its claims regarding the other patents and dedicated the patents to the public. The remaining contention was that KSR's pedal system for GM infringed claim 4 of the Engelgau patent. [***27] Teleflex has not argued that the other three claims of the patent are infringed by KSR's pedal, nor has Teleflex argued that the mechanical adjustable pedal designed by KSR for Ford infringed any of its patents.

C

The District Court granted summary judgment in KSR's favor. After reviewing the pertinent history of pedal design, the scope of the Engelgau patent, and the relevant prior art, the court considered the validity of the contested claim. By direction of *35 U.S.C. § 282*, an issued patent is presumed valid. The District Court applied *Graham*'s framework to determine whether under summary-judgment standards KSR had overcome the presumption and demonstrated that claim 4 was obvious in light of the prior art in existence when [*1738] the claimed subject matter was invented. See *§ 102(a)*.

The District Court determined, in light of the expert testimony and the parties' stipulations, that the level of ordinary skill in pedal design was "'an undergraduate degree in mechanical engineering (or an equivalent amount of industry experience) [and] familiarity with pedal control systems for vehicles.'" *298 F. Supp. 2d, at 590.* The court then set forth the [***28] relevant prior art, including the patents and pedal designs described above.

[**719] Following *Graham*'s direction, the court compared the teachings of the prior art to the claims of Engelgau. It found "little difference." *298 F. Supp. 2d, at 590.* Asano taught everything contained in claim 4 except the use of a sensor to detect the pedal's position and transmit it to the computer controlling the throttle. That additional aspect was revealed in sources such as the *'068 patent* and the sensors used by Chevrolet.

Under the controlling cases from the Court of Appeals for the Federal Circuit, however, the District Court was not permitted to stop there. The court was required also to apply the TSM test. The District Court held KSR had satisfied the test. It reasoned (1) the state of the industry would lead inevitably to combinations of electronic sensors and adjustable pedals, (2) Rixon provided the basis for these developments, and (3) Smith taught a solution to the wire chafing problems in Rixon, namely locating the sensor on the fixed structure of the pedal. This could lead to the combination of Asano, or a pedal like it, with a pedal position sensor.

The conclusion that the [***29] Engelgau design was obvious was supported, in the District Court's view, by the PTO's rejection of the broader version of claim 4. Had Engelgau included Asano in his patent application,

it reasoned, the PTO would have found claim 4 to be an obvious combination of Asano and Smith, as it had found the broader version an obvious combination of Redding and Smith. As a final matter, the District Court held that the secondary factor of Teleflex's commercial success with pedals based on Engelgau's design did not alter its conclusion. The District Court granted summary judgment for KSR.

With principal reliance on the TSM test, the Court of Appeals reversed. It ruled the District Court had not been strict enough in applying the test, having failed to make "'findings as to the specific understanding or principle within the knowledge of a skilled artisan that would have motivated one with no knowledge of [the] invention' . . . to attach an electronic control to the support bracket of the Asano assembly." *119 Fed. Appx., at 288* (brackets in original) (quoting *In re Kotzab, 217 F.3d 1365, 1371 (CA Fed. 2000)*). The Court of Appeals held that the District Court was [***30] incorrect that the nature of the problem to be solved satisfied this requirement because unless the "prior art references addressed the precise problem that the patentee was trying to solve," the problem would not motivate an inventor to look at those references. *119 Fed. Appx., at 288.*

Here, the Court of Appeals found, the Asano pedal was designed to solve the "'constant ratio problem'" -- that is, to ensure that the force required to depress the pedal is the same no matter how the pedal is adjusted -- whereas Engelgau sought to provide a simpler, smaller, cheaper adjustable electronic pedal. *Ibid.* As for Rixon, the court explained, that pedal suffered from the problem of wire chafing but was not designed to solve it. In the court's view Rixon did not teach anything helpful to Engelgau's purpose. Smith, in turn, did not relate to adjustable pedals and did not "necessarily go to the issue of motivation [*1739] to attach the electronic control on the support bracket of the pedal assembly." *Ibid.* When the patents were interpreted in this way, the Court of Appeals held, they would not have led a person of ordinary skill to put a sensor on the sort of pedal described in Asano. [***31]

[**720] That it might have been obvious to try the combination of Asano and a sensor was likewise irrelevant, in the court's view, because "' "obvious to try" has long been held not to constitute obviousness.'" *Id., at 289* (quoting *In re Deuel, 51 F.3d 1552, 1559 (CA Fed. 1995)*).

The Court of Appeals also faulted the District Court's consideration of the PTO's rejection of the broader version of claim 4. The District Court's role, the Court of Appeals explained, was not to speculate regarding what the PTO might have done had the Engelgau patent mentioned Asano. Rather, the court held, the District Court was obliged first to presume that the issued patent was valid and then to render its own independent judgment of obviousness based on a review of the prior art. The fact that the PTO had rejected the broader version of claim 4, the Court of Appeals said, had no place in that analysis.

The Court of Appeals further held that genuine issues of material fact precluded summary judgment. Teleflex had proffered statements from one expert that claim 4 "'was a simple, elegant, and novel combination of features,'" *119 Fed. Appx., at 290*, compared to Rixon, [***32] and from another expert that claim 4 was nonobvious because, unlike in Rixon, the sensor was mounted on the support bracket rather than the pedal itself. This evidence, the court concluded, sufficed to require a trial.

II

A

We begin by rejecting the rigid approach of the Court of Appeals. Throughout this Court's engagement with the question of obviousness, our cases have set forth an expansive and flexible approach inconsistent with the way the Court of Appeals applied its TSM test here. To be sure, *Graham* recognized the need for "uniformity and definiteness." *383 U.S., at 18, 86 S. Ct. 684, 15 L. Ed. 2d 545.* Yet the principles laid down in *Graham* reaffirmed the "functional approach" of *Hotchkiss, 52 U.S. 248, 11 How. 248, 13 L. Ed. 683.* See *383 U.S., at 12, 86 S. Ct. 684, 15 L. Ed. 2d 545.* To this end, *Graham* set forth a broad inquiry and invited courts, where appropriate, to look at any secondary considerations that would prove instructive. *Id., at 17, 86 S. Ct. 684, 15 L. Ed. 2d 545.*

Neither the enactment of *§ 103* nor the analysis in *Graham* disturbed this Court's earlier instructions concerning the need for caution in granting a patent based on the combination of elements found in the prior art. For over a half century, [***33] the Court has held that a "patent for a combination which only unites old elements with no change in their respective functions . . . obviously withdraws what is already known into the field of its monopoly and diminishes the resources available to skillful men." *Great Atlantic & Pacific Tea Co. v. Supermarket Equipment Corp., 340 U.S. 147, 152, 71 S. Ct. 127, 95 L. Ed. 162, 1951 Dec. Comm'r Pat. 572 (1950).* This is a principal reason for declining to allow patents for what is obvious. The combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. Three cases decided after *Graham* illustrate the application of this doctrine.

In *United States v. Adams, 383 U.S. 39, 40, 86 S. Ct. 708, 15 L. Ed. 2d 572, 174 Ct. Cl. 1293 (1966)*, a com-

panion case to *Graham*, the Court considered the obviousness of a "wet battery" that varied from [**721] prior designs in two ways: [*1740] It contained water, rather than the acids conventionally employed in storage batteries; and its electrodes were magnesium and cuprous chloride, rather than zinc and silver chloride. The Court recognized that when a patent claims a structure already known in the prior art that is altered by the mere substitution of one [***34] element for another known in the field, the combination must do more than yield a predictable result. *383 U.S., at 50-51, 86 S. Ct. 708, 15 L. Ed. 2d 572, 174 Ct. Cl. 1293.* It nevertheless rejected the Government's claim that Adams's battery was obvious. The Court relied upon the corollary principle that when the prior art teaches away from combining certain known elements, discovery of a successful means of combining them is more likely to be nonobvious. *Id., at 51-52, 86 S. Ct. 708, 15 L. Ed. 2d 572, 174 Ct. Cl. 1293.* When Adams designed his battery, the prior art warned that risks were involved in using the types of electrodes he employed. The fact that the elements worked together in an unexpected and fruitful manner supported the conclusion that Adams's design was not obvious to those skilled in the art.

In *Anderson's-Black Rock, Inc. v. Pavement Salvage Co., 396 U.S. 57, 90 S. Ct. 305, 24 L. Ed. 2d 258 (1969)*, the Court elaborated on this approach. The subject matter of the patent before the Court was a device combining two pre-existing elements: a radiant-heat burner and a paving machine. The device, the Court concluded, did not create some new synergy: The radiant-heat burner functioned just as a burner was expected to function; and the paving machine did [***35] the same. The two in combination did no more than they would in separate, sequential operation. *Id., at 60-62, 90 S. Ct. 305, 24 L. Ed. 2d 258.* In those circumstances, "while the combination of old elements performed a useful function, it added nothing to the nature and quality of the radiant-heat burner already patented," and the patent failed under *§ 103. Id., at 62, 90 S. Ct. 305, 24 L. Ed. 2d 258* (footnote omitted).

Finally, in *Sakraida v. AG Pro, Inc., 425 U.S. 273, 96 S. Ct. 1532, 47 L. Ed. 2d 784 (1976)*, the Court derived from the precedents the conclusion that when a patent "simply arranges old elements with each performing the same function it had been known to perform" and yields no more than one would expect from such an arrangement, the combination is obvious. *Id., at 282, 96 S. Ct. 1532, 47 L .Ed. 2d 784.*

The principles underlying these cases are instructive when the question is whether a patent claiming the combination of elements of prior art is obvious. When a work is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or a different one. If a person of ordinary skill can implement a predictable variation, *§ 103* likely bars its patentability. For [***36] the same reason, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill. *Sakraida* and *Anderson's-Black Rock* are illustrative -- a court must ask whether the improvement is more than the predictable use of prior art elements according to their established functions.

Following these principles may be [**722] more difficult in other cases than it is here because the claimed subject matter may involve more than the simple substitution of one known element for another or the mere application of a known technique to a piece of prior art ready for the improvement. Often, it will be necessary for a court to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having [*1741] ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue. To facilitate review, this analysis [***37] should be made explicit. See *In re Kahn, 441 F.3d 977, 988 (CA Fed. 2006)* ("Rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness"). As our precedents make clear, however, the analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ.

B

When it first established the requirement of demonstrating a teaching, suggestion, or motivation to combine known elements in order to show that the combination is obvious, the Court of Customs and Patent Appeals captured a helpful insight. See *Application of Bergel, 292 F.2d 955, 956-957, 48 C.C.P.A. 1102, 1961 Dec. Comm'r Pat. 504 (1961)*. As is clear from cases such as *Adams*, a patent composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art. Although common sense directs one to look with care at a patent application that claims as innovation [***38] the combination of two known devices according to their established functions, it can be important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the

way the claimed new invention does. This is so because inventions in most, if not all, instances rely upon building blocks long since uncovered, and claimed discoveries almost of necessity will be combinations of what, in some sense, is already known.

Helpful insights, however, need not become rigid and mandatory formulas; and when it is so applied, the TSM test is incompatible with our precedents. The obviousness analysis cannot be confined by a formalistic conception of the words teaching, suggestion, and motivation, or by overemphasis on the importance of published articles and the explicit content of issued patents. The diversity of inventive pursuits and of modern technology counsels against limiting the analysis in this way. In many fields it may be that there is little discussion of obvious techniques or combinations, and it often may be the case that market demand, rather than scientific literature, will drive design trends. Granting patent protection [***39] to advances that would occur in the ordinary course without real innovation retards progress and may, in the case of patents combining previously known elements, deprive prior inventions of their value or utility.

In the years since the Court of Customs and Patent Appeals set forth the [**723] essence of the TSM test, the Court of Appeals no doubt has applied the test in accord with these principles in many cases. There is no necessary inconsistency between the idea underlying the TSM test and the *Graham* analysis. But when a court transforms the general principle into a rigid rule that limits the obviousness inquiry, as the Court of Appeals did here, it errs.

C

The flaws in the analysis of the Court of Appeals relate for the most part to the court's narrow conception of the obviousness inquiry reflected in its application of the TSM test. In determining whether the subject matter of a patent claim is obvious, neither the particular motivation nor the avowed purpose of the [*1742] patentee controls. What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under *§ 103*. One of the ways in which a patent's subject matter can be proved obvious is [***40] by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims.

The first error of the Court of Appeals in this case was to foreclose this reasoning by holding that courts and patent examiners should look only to the problem the patentee was trying to solve. *119 Fed. Appx., at 288*. The Court of Appeals failed to recognize that the problem motivating the patentee may be only one of many addressed by the patent's subject matter. The question is not whether the combination was obvious to the patentee but whether the combination was obvious to a person with ordinary skill in the art. Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

The second error of the Court of Appeals lay in its assumption that a person of ordinary skill attempting to solve a problem will be led only to those elements of prior art designed to solve the same problem. *Ibid.* The primary purpose of Asano was solving the constant ratio problem; so, the court concluded, [***41] an inventor considering how to put a sensor on an adjustable pedal would have no reason to consider putting it on the Asano pedal. *Ibid.* Common sense teaches, however, that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle. Regardless of Asano's primary purpose, the design provided an obvious example of an adjustable pedal with a fixed pivot point; and the prior art was replete with patents indicating that a fixed pivot point was an ideal mount for a sensor. The idea that a designer hoping to make an adjustable electronic pedal would ignore Asano because Asano was designed to solve the constant ratio problem makes little sense. A person of ordinary skill is also a person of ordinary creativity, not an automaton.

The same constricted analysis led the Court of Appeals to conclude, in error, that a patent claim cannot be proved obvious merely by showing that the combination of elements was "obvious to try." *Id., at 289* (internal quotation marks omitted). When there is a design need or market pressure to solve a problem [***42] and there are a finite number of identified, predictable [**724] solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. In that instance the fact that a combination was obvious to try might show that it was obvious under *§ 103*.

The Court of Appeals, finally, drew the wrong conclusion from the risk of courts and patent examiners falling prey to hindsight bias. A factfinder should be aware, of course, of the distortion caused by hindsight bias and must be cautious of arguments reliant upon *ex post* reasoning. See *Graham, 383 U.S., at 36, 86 S. Ct. 684, 15 L. Ed. 2d 545* (warning against a "temptation to read into the prior art the teachings of the invention in issue" and instructing courts to "'guard against slipping into the use of hindsight'" (quoting *Monroe Auto Equipment Co. v. Heckethorn Mfg. & Supply Co., 332 F.2d 406, 412 (CA6 1964)))*. Rigid preventative rules that deny factfinders

recourse to common sense, however, are [*1743] neither necessary under our case law nor consistent with it.

We note the [***43] Court of Appeals has since elaborated a broader conception of the TSM test than was applied in the instant matter. See, *e.g., DyStar Textilfarben GmbH & Co. Deutschland KG v. C. H. Patrick Co., 464 F.3d 1356, 1367 (2006)* ("Our suggestion test is in actuality quite flexible and not only permits, but *requires*, consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs., Inc., 464 F.3d 1286, 1291 (2006)* ("There is flexibility in our obviousness jurisprudence because a motivation may be found *implicitly* in the prior art. We do not have a rigid test that requires an actual teaching to combine . . ."). Those decisions, of course, are not now before us and do not correct the errors of law made by the Court of Appeals in this case. The extent to which they may describe an analysis more consistent with our earlier precedents and our decision here is a matter for the Court of Appeals to consider in its future cases. What we hold is that the fundamental misunderstandings identified above led the Court of Appeals in this case to apply a test inconsistent with our patent law decisions.

III

When we apply the standards we have [***44] explained to the instant facts, claim 4 must be found obvious. We agree with and adopt the District Court's recitation of the relevant prior art and its determination of the level of ordinary skill in the field. As did the District Court, we see little difference between the teachings of Asano and Smith and the adjustable electronic pedal disclosed in claim 4 of the Engelgau patent. A person having ordinary skill in the art could have combined Asano with a pedal position sensor in a fashion encompassed by claim 4, and would have seen the benefits of doing so.

A

Teleflex argues in passing that the Asano pedal cannot be combined with a sensor in the manner described by claim 4 because of the design of Asano's pivot mechanisms. See Brief for Respondents 48-49, and n. 17. Therefore, Teleflex reasons, even if adding a sensor to Asano was obvious, that does not establish that claim 4 encompasses obvious subject matter. This argument was not, however, [**725] raised before the District Court. There Teleflex was content to assert only that the problem motivating the invention claimed by the Engelgau patent would not lead to the solution of combining of Asano with a sensor. See Teleflex's Response [***45] to KSR's Motion for Summary Judgment of Invalidity in No. 02-74586 (ED Mich.), pp. 18-20, App. 144a-146a. It is also unclear whether the current argument was raised before the Court of Appeals, where Teleflex advanced the nonspecific, conclusory contention that combining Asano with a sensor would not satisfy the limitations of claim 4. See Brief for Plaintiffs-Appellants in No. 04-1152 (CA Fed.), pp. 42-44. Teleflex's own expert declarations, moreover, do not support the point Teleflex now raises. See Declaration of Clark J. Radcliffe, Ph.D., Supplemental App. 204-207; Declaration of Timothy L. Andresen, *id.*, at 208-210. The only statement in either declaration that might bear on the argument is found in the Radcliffe declaration:

> Asano . . . and Rixon . . . are complex mechanical linkage-based devices that are expensive to produce and assemble and difficult to package. It is exactly these difficulties with prior art designs that [Engelgau] resolves. The use of an adjustable pedal with a single pivot reflecting pedal position combined with an electronic control mounted between the [*1744] support and the adjustment assembly at that pivot was a simple, elegant, and novel combination [***46] of features in the Engelgau *'565 patent*." *Id.*, at 206, P16.

Read in the context of the declaration as a whole this is best interpreted to mean that Asano could not be used to solve "the problem addressed by Engelgau '565[:] to provide a less expensive, more quickly assembled, and smaller package adjustable pedal assembly with electronic control." *Id.*, at 205, P10.

The District Court found that combining Asano with a pivot-mounted pedal position sensor fell within the scope of claim 4. *298 F. Supp. 2d, at 592-593*. Given the sigificance of that finding to the District Court's judgment, it is apparent that Teleflex would have made clearer challenges to it if it intended to preserve this claim. In light of Teleflex's failure to raise the argument in a clear fashion, and the silence of the Court of Appeals on the issue, we take the District Court's conclusion on the point to be correct.

B

The District Court was correct to conclude that, as of the time Engelgau designed the subject matter in claim 4, it was obvious to a person of ordinary skill to combine Asano with a pivot-mounted pedal position sensor. There then existed a marketplace that created a strong [***47] incentive to convert mechanical pedals to electronic pedals, and the prior art taught a number of methods for achieving this advance. The Court of Appeals considered the issue too narrowly by, in effect, asking whether a pedal designer writing on a blank slate would have cho-

sen both Asano and a modular sensor similar to the ones used in the Chevrolet truckline and disclosed in the *'068 patent*. The District Court employed this narrow inquiry as well, though it reached the correct result nevertheless. The proper question to have asked was whether a pedal designer of ordinary skill, facing the wide range of needs created by developments in the field of endeavor, [**726] would have seen a benefit to upgrading Asano with a sensor.

In automotive design, as in many other fields, the interaction of multiple components means that changing one component often requires the others to be modified as well. Technological developments made it clear that engines using computer-controlled throttles would become standard. As a result, designers might have decided to design new pedals from scratch; but they also would have had reason to make pre-existing pedals work with the new engines. Indeed, upgrading its [***48] own pre-existing model led KSR to design the pedal now accused of infringing the Engelgau patent.

For a designer starting with Asano, the question was where to attach the sensor. The consequent legal question, then, is whether a pedal designer of ordinary skill starting with Asano would have found it obvious to put the sensor on a fixed pivot point. The prior art discussed above leads us to the conclusion that attaching the sensor where both KSR and Engelgau put it would have been obvious to a person of ordinary skill.

The *'936 patent* taught the utility of putting the sensor on the pedal device, not in the engine. Smith, in turn, explained to put the sensor not on the pedal's footpad but instead on its support structure. And from the known wire-chafing problems of Rixon, and Smith's teaching that "the pedal assemblies must not precipitate any motion in the connecting wires," Smith, col. 1, lines 35-37, Supplemental App. 274, the designer would know to place the sensor on a nonmoving part of the pedal structure. The most obvious nonmoving point on the structure from which a sensor can [*1745] easily detect the pedal's position is a pivot point. The designer, accordingly, would follow Smith [***49] in mounting the sensor on a pivot, thereby designing an adjustable electronic pedal covered by claim 4.

Just as it was possible to begin with the objective to upgrade Asano to work with a computer-controlled throttle, so too was it possible to take an adjustable electronic pedal like Rixon and seek an improvement that would avoid the wire-chafing problem. Following similar steps to those just explained, a designer would learn from Smith to avoid sensor movement and would come, thereby, to Asano because Asano disclosed an adjustable pedal with a fixed pivot.

Teleflex indirectly argues that the prior art taught away from attaching a sensor to Asano because Asano in its view is bulky, complex, and expensive. The only evidence Teleflex marshals in support of this argument, however, is the Radcliffe declaration, which merely indicates that Asano would not have solved Engelgau's goal of making a small, simple, and inexpensive pedal. What the declaration does not indicate is that Asano was somehow so flawed that there was no reason to upgrade it, or pedals like it, to be compatible with modern engines. Indeed, Teleflex's own declarations refute this conclusion. Dr. Radcliffe states that [***50] Rixon suffered from the same bulk and complexity as did Asano. See *id.*, at 206. Teleflex's other expert, however, explained that Rixon was itself designed by adding a sensor to a pre-existing mechanical pedal. See *id.*, at 209. If Rixon's base pedal was not too flawed to upgrade, then Dr. Radcliffe's declaration does not show Asano was either. Teleflex may have made a plausible argument that Asano is inefficient as compared [**727] to Engelgau's preferred embodiment, but to judge Asano against Engelgau would be to engage in the very hindsight bias Teleflex rightly urges must be avoided. Accordingly, Teleflex has not shown anything in the prior art that taught away from the use of Asano.

Like the District Court, finally, we conclude Teleflex has shown no secondary factors to dislodge the determination that claim 4 is obvious. Proper application of *Graham* and our other precedents to these facts therefore leads to the conclusion that claim 4 encompassed obvious subject matter. As a result, the claim fails to meet the requirement of *§ 103*.

We need not reach the question whether the failure to disclose Asano during the prosecution of Engelgau voids the presumption of validity given [***51] to issued patents, for claim 4 is obvious despite the presumption. We nevertheless think it appropriate to note that the rationale underlying the presumption -- that the PTO, in its expertise, has approved the claim -- seems much diminished here.

IV

A separate ground the Court of Appeals gave for reversing the order for summary judgment was the existence of a dispute over an issue of material fact. We disagree with the Court of Appeals on this point as well. To the extent the court understood the *Graham* approach to exclude the possibility of summary judgment when an expert provides a conclusory affidavit addressing the question of obviousness, it misunderstood the role expert testimony plays in the analysis. In considering summary judgment on that question the district court can and should take into account expert testimony, which may resolve or keep open certain questions of fact. That is not

the end of the issue, however. The ultimate judgment of obviousness is a legal determination. *Graham, 383 U.S., at 17, 86 S. Ct. 684, 15 L. Ed. 2d 545*. Where, as here, the content of the prior art, the scope of the patent [*1746] claim, and the level of ordinary skill in the art are not in material dispute, and [***52] the obviousness of the claim is apparent in light of these factors, summary judgment is appropriate. Nothing in the declarations proffered by Teleflex prevented the District Court from reaching the careful conclusions underlying its order for summary judgment in this case.

* * *

We build and create by bringing to the tangible and palpable reality around us new works based on instinct, simple logic, ordinary inferences, extraordinary ideas, and sometimes even genius. These advances, once part of our shared knowledge, define a new threshold from which innovation starts once more. And as progress beginning from higher levels of achievement is expected in the normal course, the results of ordinary innovation are not the subject of exclusive rights under the patent laws. Were it otherwise patents might stifle, rather than promote, the progress of useful arts. See *U.S. Const., Art. I, § 8, cl. 8*. These premises led to the bar on patents claiming obvious subject matter established in *Hotchkiss* and codified in *§ 103*. Application of the bar must not be confined within a test or formulation too constrained to serve its purpose.

KSR provided convincing evidence that mounting a modular [***53] sensor on a fixed pivot point of the Asano pedal was a design step well within the [**728] grasp of a person of ordinary skill in the relevant art. Its arguments, and the record, demonstrate that claim 4 of the Engelgau patent is obvious. In rejecting the District Court's rulings, the Court of Appeals analyzed the issue in a narrow, rigid manner inconsistent with *§ 103* and our precedents. The judgment of the Court of Appeals is reversed, and the case remanded for further proceedings consistent with this opinion.

It is so ordered.

REFERENCES: Go To Full Text Opinion

Go To Supreme Court Brief(s)

Go To Supreme Court Transcripts

RELATED PROCEEDINGS APPENDIX

None.